A Semantic Metadata Enrichment Software Ecosystem (SMESE): its Prototypes for Digital Libraries, Metadata Enrichments and Assisted Literature Reviews

by

Ronald BRISEBOIS

MANUSCRIPT-BASED THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE IN PARTIAL FULFILLMENT FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
Ph.D.

MONTREAL, JUNE 20, 2017

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

BOARD OF EXAMINERS (PH.D. THESIS)

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

Mr. Alain Abran, Thesis Supervisor
Software and Information Technology Engineering Department at École de technologie supérieure

Mr. Ghyslain Gagnon, Chair of the Board of Examiners
Electrical Engineering Department at École de technologie supérieure

Mr. Alain April, Member of the Board of Examiners
Software and Information Technology Engineering Department at École de technologie supérieure

Mrs. Cherifa Mansoura Liamani, External Evaluator
Business Architect at TEKsystems

THIS THESIS WAS PRENSENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND THE PUBLIC

ON MAY 19TH 2017

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGMENT

Sincere thanks to my wife Rita Benavente for all her help, understanding, encouragement and patience; to my thesis supervisor, Dr. Alain Abran, for his time and invaluable guidance; as well as to Dr. Apollinaire Nadembéga, Philippe N'techobo and all those who helped improve the quality of this research work, day after day.

A SEMANTIC METADATA ENRICHMENT SOFTWARE ECOSYSTEM (SMESE): ITS PROTOTYPES FOR DIGITAL LIBRARIES, METADATA ENRICHMENTS AND ASSISTED LITERATURE REVIEWS

Ronald BRISEBOIS

ABSTRACT

Contribution 1: Initial design of a semantic metadata enrichment ecosystem (SMESE) for Digital Libraries

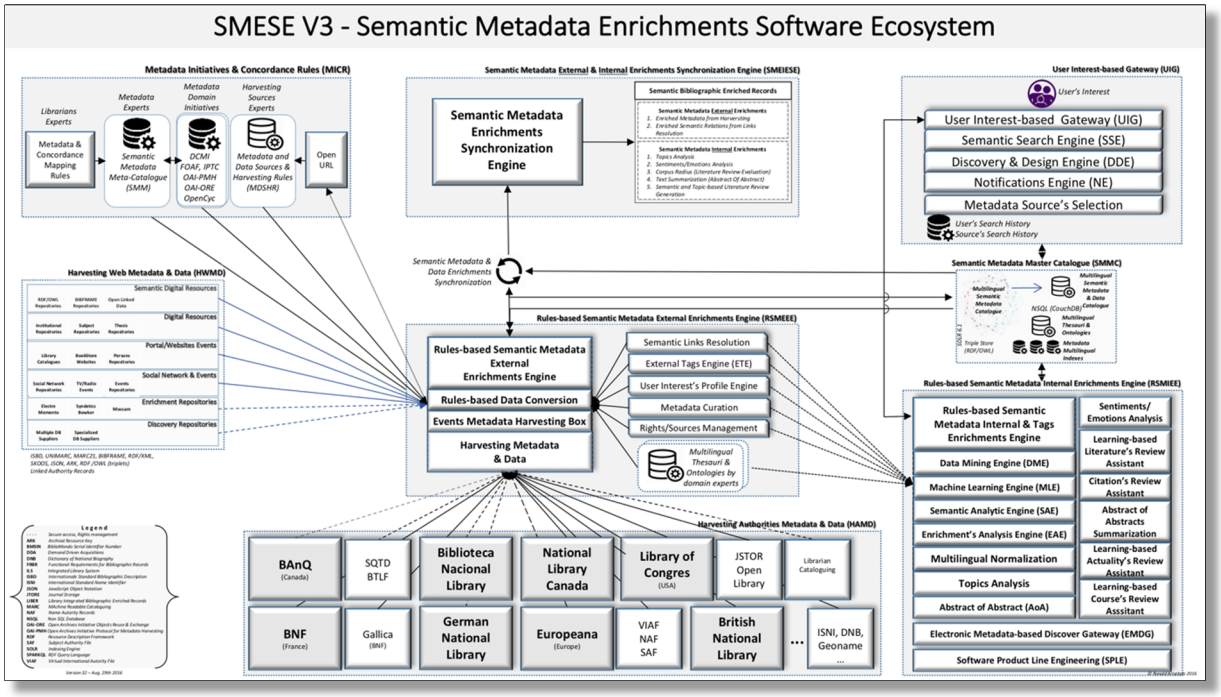
The Semantic Metadata Enrichments Software Ecosystem (SMESE V1) for Digital Libraries (DLs) proposed in this paper implements a Software Product Line Engineering (SPLE) process using a metadata-based software architecture approach. It integrates a components-based ecosystem, including metadata harvesting, text and data mining and machine learning models. SMESE V1 is based on a generic model for standardizing meta-entity metadata and a mapping ontology to support the harvesting of various types of documents and their metadata from the web, databases and linked open data. SMESE V1 supports a dynamic metadata-based configuration model using multiple thesauri.

The proposed model defines rules-based crosswalks that create pathways to different sources of data and metadata. Each pathway checks the metadata source structure and performs data and metadata harvesting. SMESE V1 proposes a metadata model in six categories of metadata instead of the four currently proposed in the literature for DLs; this makes it possible to describe content by defined entity, thus increasing usability. In addition, to tackle the issue of varying degrees of depth, the proposed metadata model describes the most elementary aspects of a harvested entity. A mapping ontology model has been prototyped in SMESE V1 to identify specific text segments based on thesauri in order to enrich content metadata with topics and emotions; this mapping ontology also allows interoperability between existing metadata models.

Contribution 2: Metadata enrichments ecosystem based on topics and interests

The second contribution extends the original SMESE V1 proposed in Contribution 1. Contribution 2 proposes a set of topic- and interest-based content semantic enrichments. The improved prototype, SMESE V3 (see following figure), uses text analysis approaches for sentiment and emotion detection and provides machine learning models to create a semantically enriched repository, thus enabling topic- and interest-based search and discovery. SMESE V3 has been designed to find short descriptions in terms of topics, sentiments and emotions. It allows efficient processing of large collections while keeping the semantic and statistical relationships that are useful for tasks such as:

1. topic detection,
2. contents classification,
3. novelty detection,
4. text summarization,
5. similarity detection.



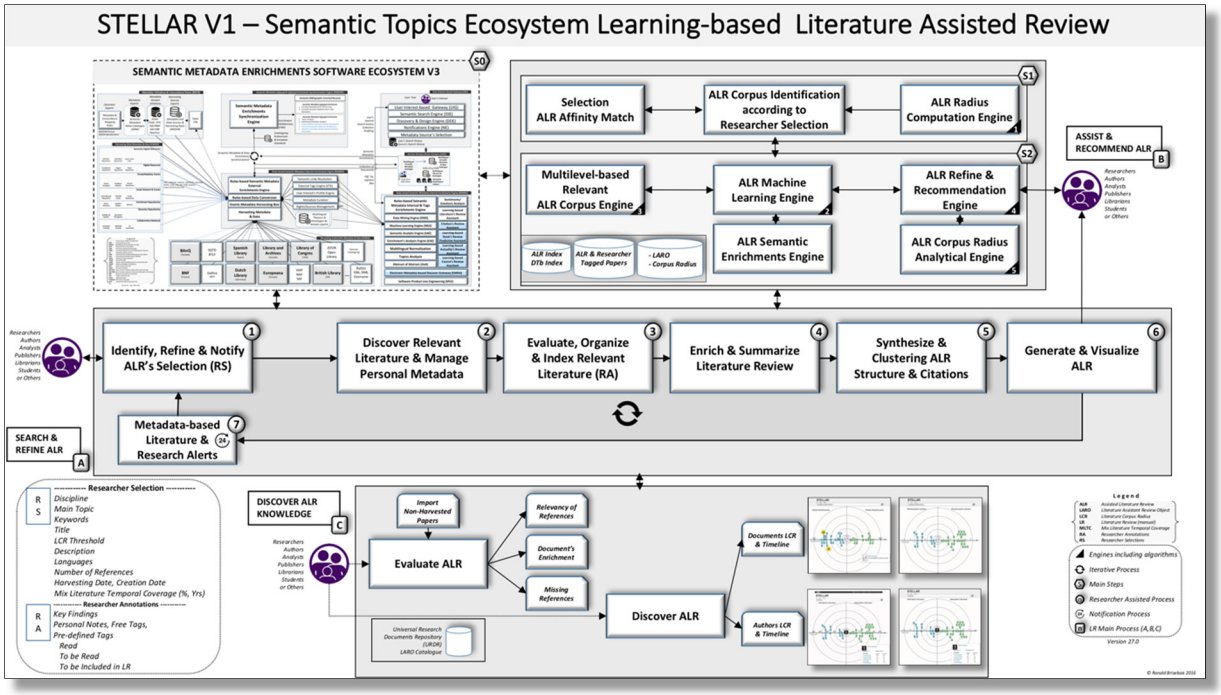
SMESE V3 – Semantic Metadata Enrichments Software Ecosystem for Digital Libraries

Contribution 3: Metadata-based scientific assisted literature review

The third contribution proposes an assisted literature review (ALR) prototype, STELLAR V1 (Semantic Topics Ecosystem Learning-based Literature Assisted Review), based on machine learning models and a semantic metadata ecosystem. Its purpose is to identify, rank and recommend relevant papers for a literature review (LR). This third prototype can assist researchers, in an iterative process, in finding, evaluating and annotating relevant papers harvested from different sources and input into the SMESE V3 platform, available at any time. The key elements and concepts of this prototype are:

1. text and data mining,
2. machine learning models,
3. classification models,
4. researchers annotations,
5. semantically enriched metadata.

STELLAR V1 helps the researcher to build a list of relevant papers according to a selection of metadata related to the subject of the ALR. The following figure presents the model, the related machine learning models and the metadata ecosystem used to assist the researcher in the task of producing an ALR on a specific topic.



STELLAR V1 – Semantic Topics Ecosystem Learning-based Literature Assisted Review

Keywords: Digital library, emotion detection, literature review, literature review enrichment, machine learning models, metadata enrichment, semantic metadata enrichment, sentiment analysis, software product line engineering, text and data mining, topic detection.

A SEMANTIC METADATA ENRICHMENT SOFTWARE ECOSYSTEM (SMESE): ITS PROTOTYPES FOR DIGITAL LIBRARIES, METADATA ENRICHMENTS AND ASSISTED LITERATURE REVIEWS

Ronald BRISEBOIS

RÉSUMÉ

Contribution 1 : Un écosystème d'enrichissements sémantiques des métadonnées (SMESE) pour des bibliothèques digitales

L'écosystème de logiciels d'enrichissements de métadonnées sémantiques (SMESE V1) proposé dans ce travail de recherche a implémenté une approche d'ingénierie de ligne de produits logiciels (SPLE) utilisant une architecture logicielle basée sur les métadonnées. Cet écosystème est basé sur le moissonnage de métadonnées, l'exploration de textes et de données et les modèles d'apprentissage automatique. SMESE V1 est basé sur un modèle générique de normalisation d'entités, de métadonnées et d'ontologies croisées capables de supporter le moissonnage de tout type de documents et de leurs métadonnées à partir du Web structuré et du Web non structuré ainsi que des données ouvertes et liées. Le design de SMESE V1 inclue un modèle de reconfiguration dynamique basé sur les métadonnées et sur plusieurs thésaurus par domaine d'application.

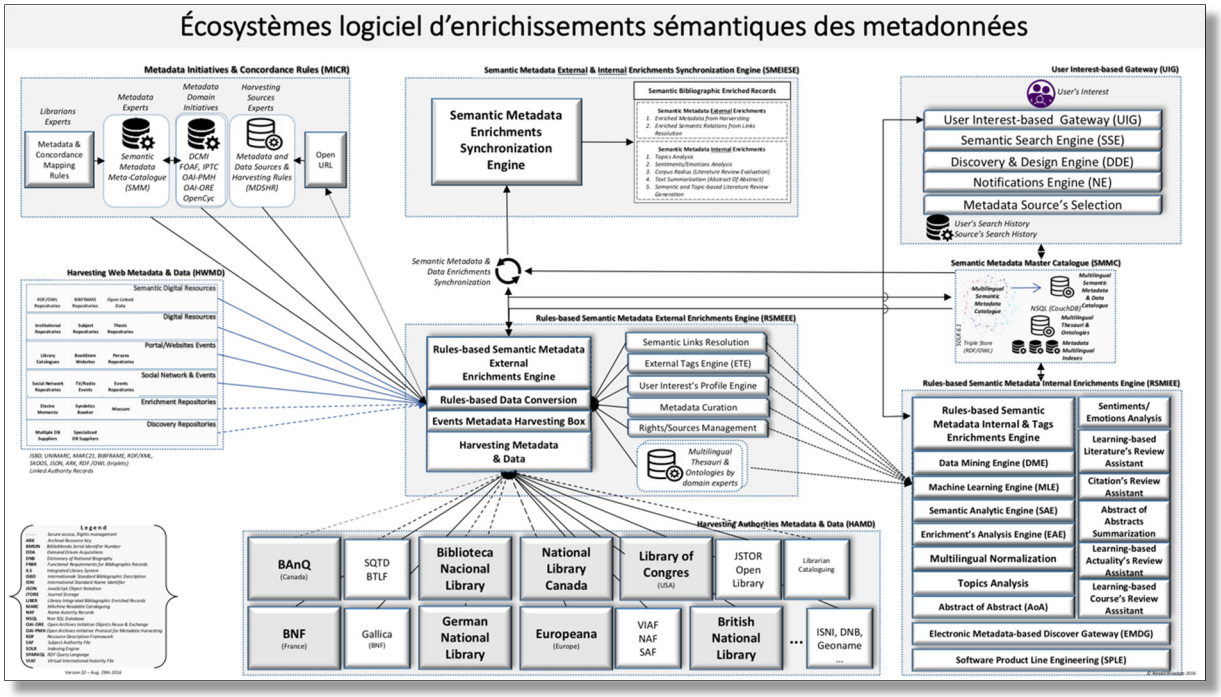
Le modèle proposé définit des règles de traduction ou de moissonnage qui créent des interfaces vers différentes sources de données et métadonnées. Chaque interface vérifie la structure de la source de métadonnées, puis effectue le moissonnage des données et des métadonnées. SMESE V1 propose un modèle de métadonnées avec six catégories de métadonnées au lieu des quatre utilisées actuellement dans la littérature afférente aux bibliothèques digitales. Ce modèle permet de mieux décrire les contenus afin d'accroitre leur utilisabilité. En plus, afin de résoudre la question des degrés de profondeur des métadonnées, le modèle de métadonnées proposé décrit les aspects les plus élémentaires d'une entité moissonnée correspondant à une structure de données. SMESE V1 inclue un modèle de mise en correspondance ontologique qui permet d'identifier des segments de texte spécifiques en utilisant des thésaurus pour enrichir les contenus de nouvelles métadonnées reliées à l'identification des sujets et des émotions. Ce

modèle de mise en correspondance ontologique permet également l'interopérabilité entre les modèles de métadonnées existants.

Contribution 2 : Un écosystème d'enrichissements métadonnées basé sur les sujets et intérêts

La contribution 2 présente une mise en œuvre améliorée de la version originale de SMESE V1, proposé dans la contribution 1 ; en effet, la contribution 2 propose des enrichissements de contenu basés sur les sujets et les intérêts. Ce prototype amélioré SMESE V3 (voir figure 1) utilise des approches d'analyse de texte pour la détection des sentiments et des émotions. Il crée un référentiel sémantique enrichi de métadonnées qui permettent la recherche et la découverte basées sur les intérêts. Il a été conçu pour trouver de courtes descriptions, en termes de sujets, de sentiments et d'émotions. Il permet un traitement efficace de grandes collections de données tout en préservant les relations sémantiques et statistiques utiles pour des tâches telles que :

1. détection de sujets,
2. classification de contenus,
3. détection de nouveautés,
4. synthèse de textes,
5. détection de similitude.



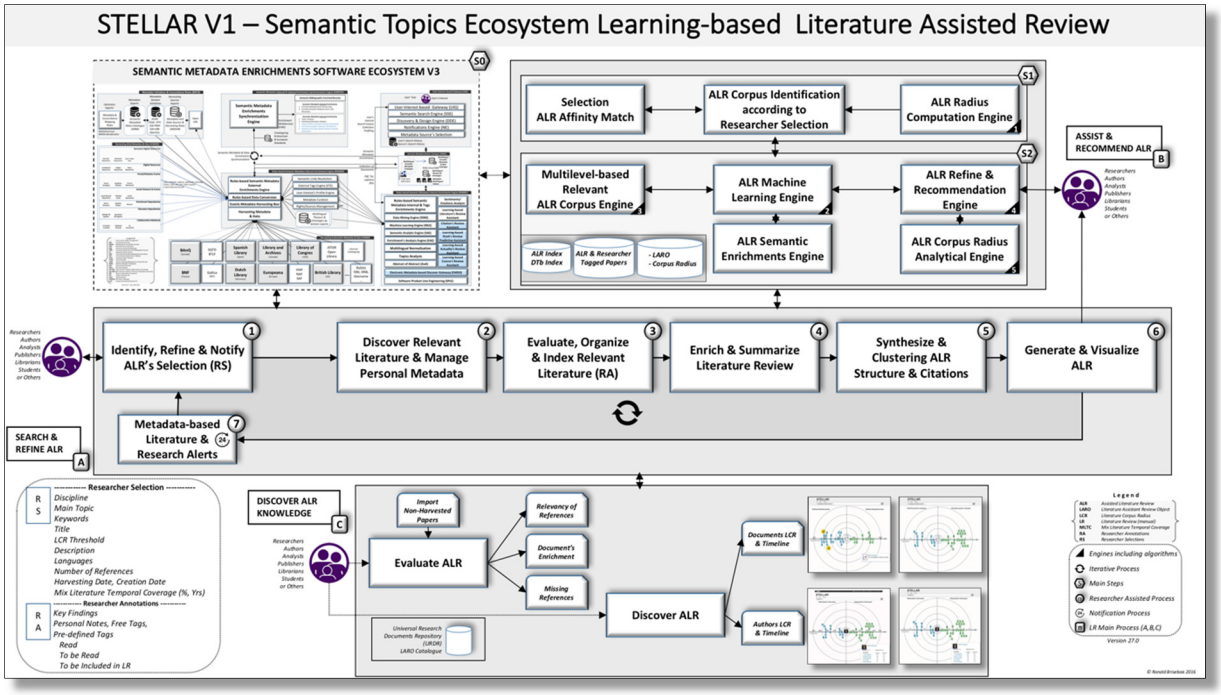
SMESE V3 – Écosystèmes logiciel d'enrichissements sémantiques des métadonnées pour bibliothèques

Contribution 3 : Une revue de littérature scientifique assistée

La contribution 3 propose un prototype (STELLAR V1- Semantic Topics Ecosystem Learning-based Literature Assisted Review V1) qui permet d'assister les chercheurs dans leurs processus de préparation d'une revue de littérature. Ce prototype de revue de littérature assistée est basé sur un écosystème de métadonnées sémantiques. Il permet d'identifier, d'évaluer et de recommander les articles scientifiques pertinents pour une revue de littérature. Le troisième prototype, STELLAR V1, permet itérativement de trouver, d'évaluer et d'annoter les articles pertinents disponibles dans la plateforme SMESE à tout moment. Les éléments et concepts clés utilisés par le prototype STELLAR V1 sont :

1. l'exploration de textes et des données,
2. les modèles d'apprentissage automatique,
3. les modèles de classification,
4. les articles annotés des chercheurs,
5. les métadonnées enrichies sémantiquement.

Ce prototype aide à identifier et à recommander les articles pertinents et leur classement lié à un sujet spécifique selon la sélection des chercheurs. La figure suivante présente le modèle, les processus associés et l'écosystème des métadonnées pour aider le chercheur dans la tâche de produire une revue de littérature reliée à un sujet spécifique.



STELLAR V1 – Écosystème sémantique d'apprentissage et d'assistance à la création de revues de littérature

Mot clés : Bibliothèque numérique, détection des émotions, revue de la littérature, enrichissement de la revue de la littérature, modèles d'apprentissage automatique, enrichissement des métadonnées, enrichissement des métadonnées sémantiques, analyse des sentiments, ingénierie des lignes de produits logiciels.

TABLE OF CONTENTS

LIST OF TABLES

Page

LIST OF FIGURES

Page

LIST OF ABREVIATIONS

AoA	Abstract of Abstracts
AKMiner	Academic Knowledge Miner
ALR	Assisted Literature Review
ALRO	Assisted Literature Review Object
ANN	Artificial Neural Network
ASE	Action Science Explorer
ATS	Automatic Text Summarization
BIBFRAME	BIBliographic FRAMEwork
BM	BiblioMondo
BNF	Bibliothèque Nationale de France
CBSD	Component-Based Software Development
CEKE	Citation Enhanced Keyphrase Extraction
COPA	Component-Oriented Platform Architecting
DC	Dublin Core
DL	Digital Libraries
DOMRM	Dynamic and Optimized Metadata-based Reconfiguration Model
DRME	Digital Resources Metadata Enrichments
DTB	Dynamic Topic-Based
EME	Entity Metadata Enrichment
LCR	Literature Corpus Radius
LDA	Latent Dirichlet Allocation
LOD	Linked Open Data
LR	Literature Review
LSA	Latent Semantic Analysis
LTM	Latent Tree Model
MARC	MAchine Readable Cataloguing
MCR	Multi-Candidate Reduction
MD	Material Design
MFD	Mobile First Design

ML	Machine Learning
MLM	Machine Learning Model
MMR	Maximal Marginal Relevance
NB	Naïve Bayes
NLP	Natural Language Processing
NMF	Nonnegative Matrix Factorization
PTRA	Paper Time Ranking Algorithm
POS	Part-Of-Speech
RA	Researcher Annotation
RDA	Resource Description and Access
RDF	Resource Description Framework
RRN	Research Relevant Novelty
RS	Researcher Selection
RUINCE	Recommended User Interest-based New Content of Events
SA	Sentiment Analysis
SEA	Sentiment & Emotion Analysis
SECO	Software Ecosystems
SIR	Semantic Information Retrieval
SME	Semantic Metadata Enrichment
SMESE	Semantic Metadata Enrichment Software Ecosystem
SML	Supervised Machine Learning
SPLE	Software Product Line Engineering
SPLE-DSP	Software Product Line Engineering – Decision Support Process
STD	Semantic Topic Detection
STELLAR	Semantic Topics Ecosystem Learning-based Literature Assisted Review
SVD	Singular Value Decomposition
SVM	Support Vector Machine
TDM	Text and Data Mining
TF-IDF	Term Frequency–Inverse Document Frequency
TSVD	Truncated Singular Value Decomposition

UML	Unsupervised Machine Learning
UNIMARC	UNIversal MAchine Readable Cataloguing
URDR	Universal Research Documents Repository
URI	Unique Resource Identifier
VR	Virtual Reality
VSM	Virtual Reality

INTRODUCTION

With more and more content, data and metadata available, understanding how users search, catalogue, rank, identify and summarize content relevant to their interests or emotions is challenging. To solve this puzzle, the semantic web approach has been explored. Indeed, there is growing research on interaction paradigms investigating how users—library users or researchers, for example—may benefit from the expressive power of the semantic web (Jeremić, Jovanović, & Gašević, 2013; Khriyenko & Nagy, 2011; Lécué et al., 2014; Ngan & Kanagasabai, 2013; Rettinger, Losch, Tresp, D'Amato, & Fanizzi, 2012). The semantic web may be defined as the transformation of the World Wide Web to a database of linked resources, where data is widely reused and shared (Lacasta, Nogueras-Iso, Falquet, Teller, & Zarazaga-Soria, 2013).

Notice that, in order to make information accessible, libraries perform several activities; one of the most fundamental is cataloguing. And in the new digital era, there is a common need, in particular for digital libraries (DLs), to be able to:

1. automate the identification and aggregation of metadata,
2. assist in the cataloguing and enrichment of content metadata.

Currently, rich information within text can be utilized to reveal meaningful semantic metadata, such as topics, sentiments, emotions and semantic relationships. The human brain has an inherent ability to identify topics, emotions and sentiments in written or spoken language. However, the Internet, social media and content repositories have expanded the number of sources, the volume of information and the number of relationships so drastically that it has become difficult for people to process all this information. It is therefore important to have high-speed computers with algorithms that can search the growing myriad of data and metadata available and extract, enrich, curate and recommend meaningful semantic metadata associated with content or events.

While computer search engines struggle to understand the meaning of natural language, semantically enriching metadata may improve those capabilities. Although there may be no relationship between the individual words of a topic or sentiment, domain thesauri do express

associative relationships between words, ontologies, entities, metadata represented as triplets.

Finding bibliographic references or semantic relationships in texts makes it possible to localize specific text segments using text data mining (TDM) and machine learning models (MLM) to enrich a set of semantic metadata.

Today, semantic web technologies, for example in DLs, offer a new level of flexibility, interoperability and a way to enhance peer communications and knowledge sharing by expanding the usefulness of the DL for searching and discovering content.

Unfortunately, to take advantage of the power of the semantic web, the poor quality of the metadata in many digital collections needs to be addressed. In the public domain there is a scarcity of search engines that follow a semantic approach to collection search and browse (Ngan & Kanagasabai, 2013).

To address these research issues, this thesis proposes a multiplatform architecture, called Semantic Metadata Enrichment Software Ecosystem (SMESE), that defines a meta-entity model and a meta-metadata model for all library materials or events in North America or Europe. SMESE is also designed to be interoperable with existing tools that use standard and non-universal models such as MAchine Readable Cataloguing (MARC), Dublin Core (DC), UNIversal MARC (UNIMARC), MARC21, Resource Description Framework/Resource Description and Access (RDF/RDA) and Bibliographic Framework (BIBFRAME).

In the meantime, the software industry has evolved to multiplatform development (including mobile phones, tablets, big screens, virtual reality (VR) and watches) based on a mix of proprietary and open-source components using heterogeneous metadata. These metadata are not always structured and organized, even though they are key to increasing the capabilities of search or discover engines. Metadata integration has emerged in software ecosystems through the software product line engineering (SPLE) process. However, metadata and enriched metadata are underused in the SPLE, as well as in systems interoperability, content enrichments and literature reviews.

Even when the metadata are well structured and universal, finding relevant content remains a

major challenge in the context of DLs; the availability of millions of content items, and millions upon millions of relationships to linked content from a growing multitude of sources (e.g., online media, social media, serial publications), makes it difficult for users to find content with a specific feature not mentioned by the content's known metadata. For example, the growing availability of a multitude of documents makes it challenging for a user to find those that are relevant to a specific need, interest or emotion. To meet this need, it becomes necessary to extract hidden metadata and to find relationships to other content, persons or events; this process is called entity metadata enrichment (EME). Several EME approaches have been proposed, most of them making use of term frequency–inverse document frequency (TF-IDF) (Niu, Zhu, Pang, & El Saddik, 2016; Salton & Buckley, 1988). This thesis focuses on sentiment analysis (SA) and semantic topic detection (STD) as an EME sub-domain.

Another research objective for the SMESE platform is to increase the findability of entities matching user interest using external references or relationships and internal (text-based) semantic metadata enrichment algorithms.

EME is also relevant to the domain of scientific research content; for example, it can define the metadata about an author's research results measurement or the relevance of a journal or paper for a specific topic. Online access to research papers plays a primordial role in the dissemination of research results through conferences and journals or through new channels such as social media. This access, combined with the evolving nature of research, creates a need to facilitate and assist researchers in the iterative process of building a Literature Review (LR) using semantic metadata. An LR is an objective, organized summary of published research relevant to the topic or area under consideration. Boote and Beile (Boote & Beile, 2005) wrote:

> "Doctoral students seeking advice on how to improve their literature reviews will find little published guidance worth heeding. Most graduate students receive little or no formal training in how to analyze and synthesize the research literature in their field, and they are unlikely to find it elsewhere"(Boote & Beile, 2005).

The field of EME that allows the ranking of scientific documents (e.g., journal papers and conference papers) is referred to as scientometrics or bibliometrics (Beel et al., 2013;

Bornmann, Stefaner, Anegón, & Mutz, 2014, 2015; Cataldi, Di Caro, & Schifanella, 2016; Dong, Johnson, & Chawla, 2016; Franceschini, Maisano, & Mastrogiacomo, 2015; Hasson, Lu, & Hassoon, 2014; Madani & Weber, 2016; Marx & Bornmann, 2016; MASIC & BEGIC, 2016; Packalen & Bhattacharya, 2015; Rúbio & Gulo, 2016; Wan & Liu, 2014; S. Wang et al., 2014; M. Zhang, Zhang, & Hu, 2015).

The literature in scientometrics also uses the following terms:

1. Journal-level metrics for publisher classification, including:
    a. Impact Factor (IF),
    b. Eigenfactor,
    c. SCImago Journal Rank,
    d. h5 index.
2. Author-level metrics for author productivity and impact measurement, including:
    a. H-index,
    b. I-10 index,
    c. G-index.

A problem with manual LR production is that it is very labor-intensive; the time researchers spend searching for and analyzing relevant literature will vary according to the subject of their research. Gall et al. (Gall, Borg, & Gall, 1996) estimate that a decent literature review for a dissertation will take between three and six months to complete. Keyword-based search is not enough to address the ambiguities of an LR. Semantic metadata, which can be extracted using text mining algorithms, allow more accurate searching and may yield better results.

The researcher has to stay aware of new related subjects and/or any relevant new articles to produce a valid LR. An LR is not simply a summary of what existing documents report about a particular topic. It has to provide an analytical overview of the significant literature published on the topic and all semantically related content. In ((Carlos & Thiago, 2015; Gulo, Rubio, Tabassum, & Prado, 2015), the authors mention that an ideal literature search would retrieve most or all relevant papers for inclusion and exclude all irrelevant papers. The sources and references have to be current and relevant, cited and formatted appropriately according to discipline and journal.

Overall, the existing research contributions in scientometrics have a number of limitations since they consider only publication count, citation count or their derivatives to measure the impact of a paper.

EME may be performed manually; the human brain has an inherent ability to detect topics, emotions, relationships and sentiments in written or spoken language, and is able to summarize various types of texts, detect content relevant to a specific topic and produce an LR. However, the Internet, social media and repositories have expanded the volume of information and the number of relationships so fast that it has become difficult to process all this information manually (Appel, Chiclana, Carter, & Fujita, 2016); hence the emergence of research on text and data mining as a way to automatically extract hidden metadata from content.

Considering these research issues in EME and the limitations of existing works, this thesis proposes new approaches that could contribute to the development of improved solutions.

The thesis consists of three technical reports corresponding to each of the three contributions:

1. A Semantic Metadata Enrichment Software Ecosystem (SMESE) Based on a Multiplatform Metadata Model for DLs;
2. A Semantic Metadata Software Ecosystem Based on Sentiment and Emotion Analysis Enrichment;
3. An Assisted Literature Review using Machine Learning Models to Build a Literature Corpus and to Recommend References using their Related Radius from this Corpus.

This thesis presents complementary information that links the three technical reports and contributions along with their prototypes and algorithms, and that also facilitates an understanding of the research approach as a whole.

The key contributions of this research have been documented in the following technical reports are presented in the Appendices I, II and III:

1. Ronald Brisebois, Alain Abran and Apollinaire Nadembega. A Semantic Metadata Enrichment Software Ecosystem (SMESE) based on a Multiplatform Metadata Model for Digital Libraries, (Appendix I);

2. Ronald Brisebois, Alain Abran, Apollinaire Nadembega, and Philippe N'techobo. A Semantic Metadata Enrichment Software Ecosystem Based on Sentiment Analysis Enrichment (SMESE V3), (Appendix II);
3. Ronald Brisebois, Alain Abran, Apollinaire Nadembega, and Philippe N'techobo. An Assisted Literature Review using Machine Learning Models to Build a Literature Corpus and to Recommend References using their Related Radius from this Corpus, (Appendix III);
4. Ronald Brisebois, Apollinaire Nadembega and Alain Abran. Real Time Software Energy Consumption Measurement in the Context of Green Software, MeGSuS, Krakow, Poland, 05–07 October 2015.

This thesis is organized as follows:

1. CHAPTER 1 provides a literature review on the current challenges in semantic metadata enrichment in terms of DL software ecosystems, semantic topic detection, sentiment and emotion analysis, scientific document ranking, scientific document text summarization and assisted literature reviews;
2. CHAPTER 2 provides an overview of the key findings and contributions of the thesis;
3. The CONCLUSION summarizes the research conducted and the research findings, including the prototypes, and proposes new avenues for future work.

The actual journal submissions are included as appendix.

CHAPTER 1

LITERATURE REVIEWS

This chapter presents a literature review on the main topics of this thesis. First, it describes the modeling of software ecosystems for DLs. Metadata enrichment approaches are then analyzed in terms of, first, text-based sentiment and emotion detection, and, secondly, Assisted Literature Reviews (ALRs) and Assisted Literature Review Objects (ALROs).

1.1 Software ecosystem model for DLs

With the proliferation of content and events in today's DL, understanding how users search and discover content has become a challenge; to tackle this challenge, DL software providers make use of metadata as content selection filters. A definition of a software ecosystem (SECO) based on the semantic analysis of data has been proposed in the literature (Christensen, Hansen, Kyng, & Manikas, 2014; Manikas & Hansen, 2013; Shinozaki, Yamamoto, & Tsuruta, 2015). Another definition from (Christensen et al., 2014; Manikas & Hansen, 2013) is the interaction of a set of actors on top of a common technological platform providing a number of software solutions or services.

There is growing agreement in the literature for the general characteristics of SECOs, including:

1. common technological platform enabling outside contributions,
2. variability-enabled architecture,
3. tool support for product derivation, as well as development processes,
4. business models involving internal and external actors (Gawer & Cusumano, 2014).

(Lettner, Angerer, Prahofer, & Grunbacher, 2014) identified ten SECO characteristics that focus on technical processes for development and evolution – see Table 1.1. However, for DLs, some additional characteristics should be taken into account, such as:

1. social network and Internet of Things integration,

2. semantic metadata internal enrichments,
3. semantic metadata external enrichments,
4. user interest-based gateways.

However, to allow SECOs to provide system adaptation capabilities, it is recommended that such adaptive characteristics be included within software product lines (SPLs) (Capilla, Bosch, Trinidad, Ruiz-Cortés, & Hinchey, 2014; Harman et al., 2014; Metzger & Pohl, 2014; Olyai & Rezaei, 2015).

The SPL approach has been recommended to organizations building applications based on a common architecture and core assets (Andrés, Camacho, & Llana, 2013; Metzger & Pohl, 2014). It is therefore highly suited to DLs.

Table 1.1 SECO characteristics
Taken from (Lettner et al., 2014)

Number	Model	Characteristics
1	SECO	Internal and external developers
2	SECO	Evaluative common technological platform
3	SECO	Controlled central part
4	SECO	Enable outside contributions and extensions
5	SECO	Variability-enabled architecture
6	SECO	Shared core assets
7	SECO	Automated and tool-supported product derivation
8	SECO	Outside contributions included in the main platform
9	SECO	Social network and IoT integration

The literature proposes a number of approaches for semantic metadata enrichment (Bontcheva, Kieniewicz, Andrews, & Wallis, 2015; Fileto, Bogorny, May, & Klein, 2015; Fileto, May, et al., 2015; Krueger, Thom, & Ertl, 2015; Kunze & Hecht, 2015); however, most authors have not focused on the enrichment model applied in the present study (Fileto, Bogorny, et al., 2015; Fileto, May, et al., 2015; Krueger et al., 2015; Kunze & Hecht, 2015).

In conclusion, the main drawbacks of SECOs based on SPL and Component-Based Software Development (CBSD) for DLs are as follows:

1. SECO-based DL software does not offer a standard and interoperable metadata model;
2. Many of the proposed SECO models do not include autonomous mechanisms to guide the self-adaptation of service compositions according to changes in the computing infrastructure;
3. There is no SECO architecture that simultaneously takes into account multiple semantic enrichment aspects;
4. Current metadata and entity enrichment models are limited to only one domain for their semantic enrichment process and therefore do not include multiple enriched metadata and entity models;
5. Current metadata and entity enrichment models link only terms and DBpedia URI.

1.2 Semantic metadata enrichments: Topics, sentiments and emotions

With the availability of millions of multiform content items and the millions upon millions of relationships that connect them, finding relevant content for a specific user interest is becoming quite difficult.

To tackle this challenge, semantic information retrieval (SIR) has been proposed; SIR is the science of searching semantically for information within databases, documents, texts, multimedia files, catalogues and the web. The current SIR approaches reduce each content item in the corpus to a vector of real numbers where each vector represents ratios of counts. Most approaches make use of TF-IDF (Niu et al., 2016; Salton & Buckley, 1988). In the TF-IDF scheme, a basic vocabulary of “words” or “terms” is chosen, then for each document in the corpus, a frequency count is calculated from the number of occurrences of each word. This yields a term-by-document matrix X whose columns contain the TF-IDF values for each of the documents in the corpus; in other words, the TF-IDF scheme reduces documents of arbitrary length to fixed-length lists of numbers.

Table 1.2 compares the most common SIR text mining tools in terms of functions: keyword extraction, classification, sentiment and emotion analysis and concept extraction.

Table 1.2 SIR models and their characteristics

SIR Model	Keyword extraction	Classification	Sentiment analysis	Emotion analysis	Concept extraction
AlchemyAPI (http://www.alchemyapi.com/)	X		X	X	X
DBpedia Spotlight (https://github.com/dbpedia-spotlight)					X
Wikimeta (https://www.w3.org/2001/sw/wiki/Wikimeta)					X
Yahoo! Content Analysis API (https://developer.yahoo.com/contentanalysis/)		X			X
Open Calais (http://www.opencalais.com/)	X	X			X
Tone Analyzer (https://tone-analyzer-demo.mybluemix.net/)			X	X	
Zemanta (http://www.zemanta.com/)					X
Receptiviti (http://www.receptiviti.ai/)			X	X	
Apache Stanbol (https://stanbol.apache.org/)					X
Bitext (https://www.bitext.com/)			X		X
Mood patrol (https://market.mashape.com/soulhackerslabs/moodpatrol-emotion-detection-from-text)				X	
Aylien (http://aylien.com/)	X	X	X		
AIDA (http://senseable.mit.edu/aida/)					X
Wikifier (http://wikifier.org/)					X
TextRazor (https://www.textrazor.com/)					X
Synesketch (http://krcadinac.com/synesketch/)				X	
Toneapi (http://toneapi.com/)			X	X	

The rest of this section presents the approaches of topic detection, sentiment and emotion analysis.

1.2.1 Semantic topic detection

Semantic topic detection (STD) within SIR helps users detect topics. It has attracted significant research in several communities in the last decade, including public opinion monitoring, decision support, emergency management and social media modeling (Hurtado, Agarwal, & Zhu, 2016; Sayyadi & Raschid, 2013).

Some examples of these advances in STD are presented in (David M. Blei, Ng, & Jordan, 2003). A topic may be defined as a set of descriptive and collocated keywords/terms. Document clustering techniques have been adopted to cluster content-similar documents and extract keywords from clustered document sets as the representation of topics. The predominant method for topic detection is the latent Dirichlet allocation (LDA) (David M. Blei

et al., 2003); LDA-based approaches assume a generating process for the documents. LDA has been proven powerful because of its ability to mine semantic information from text data.

STD was designed for large and noisy data collections such as social media, and addresses both scalability and accuracy challenges. One challenge is to rapidly filter noisy and irrelevant documents, while at the same time accurately clustering and ordering a large collection.

Several approaches are proposed in the literature for text-based topic detection:

1. Short texts (Cigarrán, Castellanos, & García-Serrano, 2016; Cotelo, Cruz, Enríquez, & Troyano, 2016; Dang, Gao, & Zhou, 2016; Hashimoto, Kuboyama, & Chakraborty, 2015) such as tweets or Facebook posts;
2. Long texts (David M. Blei et al., 2003; Bougiatiotis & Giannakopoulos, 2016; P. Chen, Zhang, Liu, Poon, & Chen, 2016; Salatino & Motta, 2016; Sayyadi & Raschid, 2013; C. Zhang, Wang, Cao, Wang, & Xu, 2016) such as books, papers or documents.

In the context of this thesis, the focus is on long-text-based topic detection. (Bijalwan, Kumar, Kumari, & Pascual, 2014) conducted experiments on text and document mining; they concluded that k-nearest neighbors (KNN) provided better accuracy than naive Bayes and term-graph. The drawback of KNN is that it is quite slow.

Recently, researchers have proposed topic detection approaches using a number of information extraction techniques (IETs), such as lexicon, sliding window and boundary. Many of these techniques (P. Chen et al., 2016; Salatino & Motta, 2016; Sayyadi & Raschid, 2013; C. Zhang et al., 2016) rely heavily on simple keyword extraction from text.

One approach for topic detection, KeyGraph, was proposed in (Sayyadi & Raschid, 2013) and was inspired by the keyword co-occurrence graph and efficient graph analysis methods. KeyGraph is based on the similarity of keywords extracted from text. There are limitations to this approach, however, and it requires improvement in two respects:

1. It underestimates the leverage of the semantic information derived from topic models;

2. It measures co-occurrence relations from an isolated term-term perspective: that is, the measurement is limited to the term itself and the information context is overlooked, which can make it impossible to measure latent co-occurrence relations.

(Salatino & Motta, 2016) suggest that it is possible to forecast the emergence of novel research topics even at an early stage and to demonstrate that such an emergence can be anticipated by analyzing the dynamics of pre-existing topics. They present a method that integrates statistics and semantics for assessing the dynamics of a topic graph. Unfortunately, their approach is not fully semantic.

(P. Chen et al., 2016) propose a novel method for hierarchical topic detection where topics are obtained by clustering documents in multiple ways. They use a class of graphical models called hierarchical latent tree models (HLTMs). However, their approach is not semantic and does not consider the domain knowledge of the analyzed text.

(Hurtado et al., 2016) propose an approach that uses sentence-level association rule mining to discover topics from documents. Their method considers each sentence as a transaction and keywords within the sentence as items in the transaction. By exploring keywords (frequently co-occurring) as patterns, their method preserves contextual information in the topic mining process. Their approach is limited to keyword counting; the semantic aspect of these keywords is not taken into account.

(C. Zhang et al., 2016) propose LDA-IG, an extension of KeyGraph (Sayyadi & Raschid, 2013). It is a hybrid analysis approach integrating semantic relations and co-occurrence relations for topic detection. Specifically, their approach fuses multiple types of relations into a uniform term graph by combining idea discovery theory with a topic modeling method. These authors used a semantic relation extraction approach based on LDA that enriches the graph with semantic information. However, their approach does not include MLM, which would allow the framework itself to find new topics.

The Table 1.3 presents an overview of some recent and relevant studies on topic detection. It can be clearly observed that semantic aspect, topic correlation and machine learning techniques are not considered.

Table 1.3 Overview of work on topic detection

Works	Text size	Approaches	Semantic	Topic correlation	Machine Learning
(Dang et al., 2016)	short	Dynamic Bayesian networks	No	No	No
(Cigarrán et al., 2016)	short	Formal concept analysis (FCA)	No	No	No
(Sayyadi & Raschid, 2013)	long	Graph analysis methods	No	No	No
(Salatino & Motta, 2016)	long	Graph analysis methods	No	No	No
(P. Chen et al., 2016)	long	Probabilistic and graph analysis methods	No	No	No
(Hurtado et al., 2016)	long	Sentence-level association rule mining	No	No	No
(C. Zhang et al., 2016)	long	Probabilistic and graph analysis methods	No	No	No

To sum up this literature review, the main drawbacks of existing approaches to topic detection are as follows:

1. They are based on simple keyword extraction from text and lack semantic information that is important for understanding the document. To tackle this limitation, the present study has used semantic annotations to improve document comprehension time;
2. Co-occurrence relations across the document are commonly neglected, which leads to incomplete detection of information. Current topic modeling methods do not explicitly consider word co-occurrences. Extending topic modeling to include co-occurrence can be a computational challenge. The graph analytical approach to this extension was only an approximation that merely took into account co-occurrence information while ignoring semantic information. How to combine semantic relations and co-occurrence relations to complement each other remains a challenge;
3. Existing approaches focus on detecting prominent or distinct topics based on explicit semantic relations or frequent co-occurrence relations; as a result, they ignore latent co-occurrence relations. In other words, latent co-occurrence relations between two terms cannot be measured from an isolated term-term perspective. The context of the term needs to be taken into account;
4. More importantly, even though existing approaches take into account semantic relations, they do not include machine learning to find new topics automatically;

5. The main conclusion is that most of the studies are limited to simulations using existing algorithms. None of them contribute improvements to help detect topics more accurately.

1.2.2 Sentiment and emotion analysis

Today, many websites offer reviews of items like books, events, music, or games. TV shows and movies where the products are described and evaluated as good/bad, liked/disliked. Unfortunately, such ratings do not help users make decisions according to their own interests. With the rapid spread of social media, it has become necessary to categorize these reviews in an automated way (Niu et al., 2016); that is the objective of sentiment and emotion analysis. These analyses establish the attitude of a given person with regard to sentences, paragraphs, chapters or documents.

Note that sentiment and emotion analysis may be defined as a type of automatic classification represented by a facet. As such, there are different analysis techniques, such as keyword spotting, lexical affinity and statistical methods. However, the most commonly applied techniques belong either to the category of text classification supervised machine learning, which uses methods like naive Bayes, maximum entropy or support vector machine, or to the category of text classification unsupervised machine learning.

In this section the concepts of emotion and sentiment are used together. Emotions are also associated with mood, temperament, personality, outlook and motivation (Li & Xu, 2014; Munezero, Montero, Sutinen, & Pajunen, 2014; Shivhare & Khethawat, 2012). Indeed, the concepts of emotion and sentiment have often been used interchangeably, mostly because both refer to experiences that result from combined biological, cognitive and social influences.

According to (Balazs & Velásquez, 2016), the sentiment and emotion analysis process typically consists of a series of steps:

1. corpus or data acquisition,
2. text preprocessing,

3. opinion mining core process,
4. aggregation and summarization of results,
5. visualization.

A number of algorithms or approaches are used in the literature to perform text mining in the sentiment and emotion analysis process based on the associated document's classification:

1. Latent Dirichlet allocation (LDA) (David M. Blei et al., 2003),
2. TF-IDF (Niu et al., 2016; Salton & Buckley, 1988),
3. Latent Semantic Analysis (LSA) (Dumais, 2004),
4. Formal concept analysis (FCA) (Cigarrán et al., 2016),
5. Latent Tree Model (LTM) (P. Chen et al., 2016),
6. Naive Bayes (NB) (Moraes, Valiati, & Gavião Neto, 2013),
7. Support Vector Machine method (SVM) (Moraes et al., 2013),
8. Artificial Neural Network (ANN) (Ghiassi, Skinner, & Zimbra, 2013).

For example, Moraes et al. (Moraes et al., 2013) compare popular machine learning approaches (SVM and NB) with an ANN-based method for document-level sentiment classification. Their experimental results show that, for book datasets, SVM outperformed ANN when the number of terms exceeded 3,000. Although SVM required less training time, it needed more running time than ANN; indeed, for 3,000 terms, ANN required 15 sec training time (with negligible running time) while SVM training time was negligible (1.75 sec). As in (Moraes et al., 2013), S. Poria et al. (Poria, Cambria, Hussain, & Huang, 2015) experimented with existing approaches and showed that SVM is a better approach for text-based emotion detection.

According to (Shivhare & Khethawat, 2012), there are three main techniques for sentiment analysis:

1. *Keyword spotting* consists in developing a list of keywords—usually positive or negative adjectives—that relate to a certain sentiment. This technique classifies text by affect categories based on the presence of unambiguous affect words such as happy, sad, afraid and bored;

2. *Lexical affinity* assigns to arbitrary words a probabilistic 'affinity' for a particular emotion. The polarity of each word is determined using different unsupervised techniques. Next, it aggregates the word scores to obtain the polarity score of the text;
3. *Statistical/Learning based methods* are supervised approaches, such as Bayesian inference and support vector machines, in which a labeled corpus is used to train a classification method that builds a classification model used for predicting the polarity of novel texts. By feeding a large training corpus of affectively annotated texts into a machine learning algorithm, it is possible for the system to not only learn the affective valence of affect keywords (as in the keyword spotting approach), but also to take into account the valence of other arbitrary keywords (like lexical affinity), punctuation and word co-occurrence frequencies.

Sentiment and emotion analysis can be carried out at different levels of text granularity:

1. document (Bosco, Patti, & Bolioli, 2013; Cho, Kim, Lee, & Lee, 2014; Kontopoulos, Berberidis, Dergiades, & Bassiliades, 2013; Lin, He, Everson, & Ruger, 2012; Moraes et al., 2013; Moreo, Romero, Castro, & Zurita, 2012),
2. sentence (Abdul-Mageed, Diab, & Kübler, 2014; Appel et al., 2016; Desmet & Hoste, 2013; Niu et al., 2016; Patel & Madia, 2016),
3. phrase or clause (Tan, Na, Theng, & Chang, 2012),
4. word (L. Chen, Qi, & Wang, 2012; Ghiassi et al., 2013; Quan & Ren, 2014).

Most of the current text-based sentiment and emotion analysis approaches focus on 'optimistic', 'depressed' and 'irritated', which are difficult to identify in the text due to the following challenges:

1. ambiguity of keyword definitions,
2. inability to recognize sentences without keyword,
3. difficulty determining emotion indicators.

A number of studies have proposed sentiment and emotion analysis techniques; for example, Cho et al. (Cho et al., 2014) propose a method to improve the positive vs. negative classification performance of product reviews by merging, removing and switching the entry words of the multiple sentiment dictionaries. However, their contribution is limited to

development of a novel method of removing and switching the content of the existing sentiment lexicons.

Bao et al. (Bao et al., 2012) present an emotion-topic model, proposing to explore the connection between the evoked emotions of readers and news headlines by generating a word-emotion mapping dictionary. For each word w in the corpus, it assigns a weight for each emotion e; i.e., P(e|w) is the averaged emotion score observed in each news headline H in which w appears.

Lei et al. (Lei, Rao, Li, Quan, & Wenyin, 2014) adopt the lexicon-based approach in building the social emotion detection system for online news based on modules of document selection, part-of-speech (POS) tagging, and social emotion lexicon generation. Specifically, given the training set T and its feature set F, an emotion lexicon is generated as a V×E matrix where the (j,k) item in the matrix is the score (probability) of emotion e_k conditioned on feature f_j. Unfortunately, these authors do not explain how they extracted the features from the document.

Anusha and Sandhya (Anusha & Sandhya, 2015) propose a system for text-based emotion detection which uses a combination of machine learning and natural language processing. Their approach recognizes affect in the form of six basic emotions proposed by Ekman; they made use of the Stanford CoreNLP toolkit to create the dependency tree based on word relationships. Next, they performed phrase selection using the rules on dependency relationships that gives priority to the semantic information for the classification of a sentence's emotion. Their approach is based on the sentence.

Cambria et al. (Cambria, Gastaldo, Bisio, & Zunino, 2015) explore how the high generalization performance, low computational complexity, and fast learning speed of extreme learning machines can be exploited to perform analogical reasoning in a vector space model of affective common-sense knowledge. After performing truncated singular value decomposition (TSVD) on AffectNet, they use the Frobenius norm to derive a new matrix. For the emotion categorization model, they use the Duchenne smile and the TSVD model.

Table 1.4 presents an overview of sentiment and emotion analysis studies organized by different approaches.

Table 1.4 Overview of studies on sentiment and emotion analysis

Works	Text granularity	Approaches	Semantic	Valence	Emotion
(Cho et al., 2014)	Document	Keyword spotting		X	
(Bao et al., 2012)	Document	Statistical/Learning based methods	X		X
(Lei et al., 2014)	Phrase or clause	Lexical affinity			X
(Anusha & Sandhya, 2015)	Document	Statistical/Learning based methods	X		X
(Cambria et al., 2015)	Document	Statistical/Learning based methods	X		X

The work on sentiment and emotion analysis can be summarized as follows:

1. Traditional SA methods mainly use terms along with their frequency and part of speech, as well as rules of opinions and sentiment shifters. Semantic information is ignored in term selection, and it is difficult to find complete rules;
2. Most of the recent contributions are limited to SA elaborated in terms of positive or negative opinion and do not include analysis of emotion;
3. Existing approaches do not allow human input, which would improve accuracy;
4. Existing approaches do not combine sentiment and emotion analysis;
5. Lexicon- and ontology-based approaches provide good accuracy for text-based sentiment and emotion analysis when applying SVM techniques. In the present approach, it is more interesting to take the entire collection into account when identifying the sentiment and emotion of a book. For example, assuming that book A has 90% fear and 80% sadness while book B has 40% fear as its predominant emotion, can it be said that fear is the emotion of book B as well as book A?
6. Existing approaches do not take document collections into account. In terms of granularity, most approaches are sentence-based;
7. Existing approaches do not take sentence context into account and consequently risk losing the real emotion.

As a general conclusion to the literature review on topic detection, sentiment and emotion analysis, 95% of studies have focused on document features (e.g., sentence length, capitalized

words, document title, term frequency and sentence position) to perform text mining and have generally made use of existing algorithms or approaches (e.g., LDA, TF-IDF, LSA, TextRank, PageRank, LexRank, SVM, NB and ANN) based on features associated with the documents.

1.3 Semantic metadata enrichments based on assisted literature review objects (ALROs)

This sub-section presents several facets about assisted literature review that should be addressed:

1. scientific paper ranking,
2. text and data mining, and more specifically:
    a. automatic text summarization (ATS),
    b. scientific paper summarization,
3. automatic multi-document summarization for a literature review.

1.3.1 Scientific paper ranking

Researchers and other users discover, analyze and maintain updated bibliographies for specific research fields; this is an important phase in the production of an LR.

A number of ranking algorithms are proposed in the literature. Ranking algorithms are the procedure that search engines use to give priority and relevancy query results. Recent years have seen wider adoption of scientometric techniques for assessing the impact of publications, researchers, institutions and venues. To date, the field of scientometrics has focused on analyzing the quantitative aspects of the generation, propagation and utilization of scientific information.

Two means of measuring scientific research output are discussed in the literature: peer-review and citation-based bibliometric indicators. The main limitation of peer-review-based approaches is the subjectivity of evaluators, while citation-based approaches have been

criticized for limiting their scope to academia and neglecting the broader societal impact of research (Marx & Bornmann, 2016).

Marx and Bornmann (Marx & Bornmann, 2016) present an overview of methods based on cited references and examples of some empirical results from studies. According to the authors, it is possible to measure the target-oriented impact in specific research areas (i.e. limited to those areas) of the citation. For the authors, cited reference analysis indicates the potential of the data source. They also mention a new method known as citing side normalization, where each individual citation receives a field-specific weighting computed by dividing thecitation by the number of references in the citing work.

The literature presents other approaches for ranking scientific articles and measuring their impact (Beel et al., 2013; Bornmann et al., 2014, 2015; Cataldi et al., 2016; Dong et al., 2016; Franceschini et al., 2015; Hasson et al., 2014; Madani & Weber, 2016; Marx & Bornmann, 2016; MASIC & BEGIC, 2016; Packalen & Bhattacharya, 2015; Rúbio & Gulo, 2016; S. Wang et al., 2014; M. Zhang et al., 2015). Some approaches focus on journal ranking (Packalen & Bhattacharya, 2015), others on university and research institute ranking (Bornmann et al., 2015). However, most of these approaches consider only publication count or focus on citation analysis (citation-based approaches); the aggregate citation statistics are used to come up with evaluative metrics for measuring scientific impact. They ignore the quality of articles in terms of new contribution and scientific impact, and limit the evaluation to the quantitative aspect.

Despite several criticisms of citation-based impact measurements, it is still the subject of much scientometric research; a highly cited paper in a given scientific research field has influenced many other researchers. The main approach for scientific article ranking is citation analysis, which is essentially the number of times a paper has been cited; however, this traditional approach does not consider the publisher, conference or workshop relevance, or the possible societal impacts of a study. Furthermore, in measuring the quality of an article, peer reviews should be taken into account, as the opinion of the scientific community in that research field may help identify relevant articles. Most approaches reduce a citation to a single edge between the citing paper and the cited paper, and treat all edges equally.

Some works in scientific impact evaluation (Bornmann et al., 2014, 2015; Cataldi et al., 2016; M. Zhang et al., 2015) have focused on the ranking of universities, institutions and research teams. For instance, (M. Zhang et al., 2015) propose a comprehensive method to discover and rank collaborative research teams based on social network analysis along with traditional citation analysis and bibliometric. In their approach, research teams are ranked using indexes which include both scientific research outcomes and the closeness of co-author networks.

Their evaluation system consists of three indexes with sub-levels:

1. Team output, with four sub-levels:
    a. total quantity published,
    b. average quantity published,
    c. total quantity published in cooperation,
    d. average quantity published in cooperation.
2. Team influence, with two sub-levels:
    a. total citations,
    b. average citations.
3. Closeness of cooperation, with three sub-levels:
    a. density,
    b. network efficiency,
    c. clustering coefficient.

And for each index, they assign a weight based on the scores of 30 experts. The main drawback of their approach is the manual contributions of the experts.

Bornmann et al. (Bornmann et al., 2014, 2015) measure the performance of research institutes based on the best paper rate and the best journal rate. Best paper rate is the proportion of institutional publications that belong to the 10% most frequently cited publications in their subject area and publication year. Best journal rate is the proportion of publications that an institution publishes in the most influential journals worldwide. Unfortunately, ranking researchers, journals and institutions does not give any idea of a scientific paper's relevancy. It may nonetheless be used to compute the paper's relevancy index.

Wan and Liu (Wan & Liu, 2014) propose citation-based analysis to evaluate scientific impact of researchers expressed as an author-level Metric called the WL-index. They raise the issue of considering the number of times a cited paper is mentioned in a citing paper. According to the authors, counting based on binary citation relationships is not appropriate; indeed, in a given article, some cited references appear only once, but others appear more than once. In other words, the WL-index, a variant of the h-index, factors in the number of times a cited paper is mentioned.

Hasson et al. (Hasson et al., 2014) propose an algorithm called the Paper Time Ranking Algorithm (PTRA), which depends on three factors to rank its results: paper age, citation index and publication venue. Specifically, they give priority to each one of these parameters; for a given paper, they compute its weight as the sum of the conference or journal's impact facto, the number of citations and the age of the paper.

Rúbio and Gulo (Rúbio & Gulo, 2016) apply an MLM called ID3 to determine a paper's relevancy classification based on specialist annotations. They combine text mining efforts and bibliometric measures to automatically classify relevant papers. They make use of metadata such as year of publication, citation number, reference number and type of publication.

Madani and Weber (Madani & Weber, 2016) propose an approach that applies bibliometric analysis and keyword-based network analysis to recognize important papers. To find the most relevant papers, they apply 'eigenvector centrality'. For the patent evaluation they extracted keywords from abstracts and created a keyword-based network that was analyzed by cluster analysis to find groups of keywords making use of the minimum spanning tree method.

Wang et al (S. Wang et al., 2014) propose a unified ranking model, called MRFRank, that utilizes the mutual reinforcement relationships across networks of papers, authors and text features. More specifically, MRFRank incorporates the text features extracted and the weighted graphs constructed. For a given sentence, it extracts words and co-occurrences from the title and abstract. Next, it computes the TF-IDF of each word as the weight of this word. The main limitation of this approach is that only the abstract is used to compute the weight of a word.

Gulo et al. (Gulo et al., 2015) propose a solution that combines text mining and MLM to identify the most relevant scientific papers. Based on previous samples manually classified by domain experts, they apply a Naive Bayes Classifier to get predicted articles.

Based on this analysis of existing approaches to scientific paper ranking, a number of limitations have been identified:

1. Most existing approaches focus on the researcher's or journal's index to evaluate the impact of a research paper, ignoring the paper's index;
2. Most approaches that focus on the paper's index use only the citations count; in addition, they do not consider the paper's age, penalizing the recent papers;
3. As for the few approaches focusing on the evaluation of the paper itself, they do not take into account the social-level metric, and they do not consider the category or polarity of citations;
4. Some approaches make use of journal information to rank papers; while this is a step in the right direction, they do not consider other types of venues, such as conferences and workshops;
5. Several approaches make use of machine learning; however, they require a large manual contribution by specialists or experts to train the learning model;
6. Very few works focus on text-based analysis to identify relevant papers; those that do, limit the analysis to title and abstract.

In summary, no approach currently takes into account all these aspects of scientific papers:

1. venue age,
2. venue type,
3. venue impact,
4. year of publication,
5. number of citations,
6. citation category,
7. references,
8. author's impact,
9. author's institutes,

10. citing document of cited document.

1.3.2 Text and data mining

Text and data mining (TDM) can be defined as the automated processing of large amounts of structured digital textual content, for purposes of information retrieval, extraction, interpretation and analysis. When large amounts of data are accumulated, automated or semi-automated analysis of their content reveals patterns that allow the establishment of fact patterns invisible to the naked eye (Okerson, 2013).

There are many reasons researchers might want to utilize TDM in their research. Clark (Clark, 2013) suggests that, given the enormous growth in the volume of literature produced, researchers should apply text mining techniques to enrich their content and perform systematic literature reviews. Mining should be deployed to enhance indexing, create relevant links and improve the reading experience. In the context of TDM, text mining is a subfield of data mining that seeks to extract valuable new information from unstructured (or semi-structured) sources. It then aggregates the extracted pieces over the entire collection of source documents to uncover or derive new information. This is the preferred view that allows one to distinguish text mining from natural language processing (NLP).

ATS approaches need to produce a concise and fluent summary conveying the key information in the input (Saggion & Poibeau, 2013). Basic approaches of ATS first extract the topics discussed in the input document; then, based on these topics, sentences in the input document are scored for importance.

There are two types of summarization, depending on the input: single document summarization and multi-document summarization (Saggion & Poibeau, 2013; D. Wang, Zhu, Li, & Gong, 2013). In (D. Wang et al., 2013), Wang et al. discuss in detail the following extractive summarization methods are discussed in detail:

1. centroid-based methods,
2. graph-based methods,

3. Latent Semantic Analysis (LSA),
4. Nonnegative Matrix Factorization (NMF).

Within the context of scientific research, documents (such as journal articles, white papers, conference proceedings or research papers) have a specific organization and features that differentiate them from other types of documents such as narrative texts (R. Zhang, Li, Liu, & Gao, 2016), where the characters are very important, and factual texts, where the summarizer has to select the most important facts and present them in a sensible order while avoiding repetition (Carenini, Cheung, & Pauls, 2013). In addition, scientific papers contain certain stock expressions and sentences.

Conventional text summarization approaches are therefore inadequate for scientific paper summarization; however, such approaches may be extended and adapted. For this reason, this sub-section of related works about TDM focuses on:

1. automatic text summarization,
2. scientific paper summarization.

1.3.2.1 Automatic text summarization

According to (Saggion & Poibeau, 2013), there are two main types of automatic text summarization (ATS):

1. Extractive summarization selects the important sentences from the original input documents to form a summary;
2. Abstractive summarization (Genest & Lapalme, 2012; Gerani, Mehdad, Carenini, Ng, & Neja, 2014) paraphrases the corpus using novel sentences; this usually involves information fusion, sentence compression and reformulation. Although an abstractive summary could be more concise, it requires deep NLP techniques.

Extractive summaries are therefore more feasible and practical, and are hence the main focus in this related works section.

For extractive summarization, three approaches are presented in the literature:

1. Word scoring, in which scores are assigned to the most important words;
2. Sentence scoring, in which sentence features such as position in the document, similarity to the title, etc. are examined;
3. Graph scoring, in which relationships between sentences are analyzed.

According to (Ferreira et al., 2013), sentence scoring is the technique most widely used for extractive text summarization.

Several works on ATS are reported in the literature. Hasan and Ng (Hasan & Ng, 2014) mention that in a structured document, there are certain locations where key sentences are most likely to appear; for instance, in the abstract and the introduction. These authors claim that the lack of structural consistency in other types of structured documents, such as books, may render structural information less useful.

He et al. (Z. He et al., 2015) propose an unsupervised summarization framework from the perspective of data reconstruction. They argue that a good summary should consist of those sentences that can best reconstruct the original document. Specifically, after stemming and stop-word elimination, they break the document down into individual sentences and create a weighted term-frequency vector for every sentence; all the sentences in the document form the candidate set. Then, they find an optimal set of representative sentences to approximate the entire document, by minimizing the reconstruction error. In their approach, these authors make use of a set of summaries, obtained through a complex procedure, as input.

Fang et al. (Fang et al., 2015) present an ATS approach based on topic factors. They define topic factors as various characteristics for the description of topics; for example, capitalized words are usually the entity (organization name) and long sentences are preferred for highly technical expert documents. Since it is unfeasible to explicitly define topic factors, they introduce a latent variable to capture the implicit topic factors. In other words, for a given topic, they identify a set of factors that characterize all documents on this topic. The drawback of their approach is that it is strongly linked to topic detection; however, the authors do not propose a topic detection mechanism to support their topic aspect-oriented approach.

Dokun and Celebi (CELEBI & DOKUN, 2015) propose two approaches based on Latent Semantic Analysis (LSA) for English documents. They convert the input document to a sentence–term matrix and process it through an algorithm called Singular Value Decomposition (SVD), designed to find and model the relationships between words and sentences while reducing noise. The authors do not propose a new contribution, but only apply an existing LSA approach.

Premjith et al. (Premjith, John, & Wilscy, 2015) present an extractive summarization system that selects salient sentences from the input documents; they consider ATS as an optimization problem. First, these authors use a variant form of the Simple Matching Coefficient scheme to reduce the dimensionality of a set of sentences from input documents to be considered for summarization; next, they use the Vector Space Model (VSM) method and bag-of-words approach to represent sentences in the input documents matrix. After preprocessing the documents, they score the sentences based on features such as Term Frequency Inverse Sentence Frequency (TF-ISF) in order to aggregate cross-sentence similarity, title similarity and sentence length.

For the optimization, they define two objective functions: function 1 checks only the similarity between the centroid concepts in both the summary and the document set, and diversity of sentences in the summary; function 2 introduces semantic coverage of the sentences in the candidate summaries based on the LSA approach. The main drawback is the complexity due to the repetition of the process of objective functions.

Sankarasubramaniam et al. (Sankarasubramaniam, Ramanathan, & Ghosh, 2014) present an approach that makes use of Wikipedia and graph-based ranking. Specifically, these authors construct a bipartite sentence–concept graph, where the concepts represent Wikipedia article titles that are closest to the input sentences, and then rank the sentences for potential inclusion in a summary. Unfortunately, these authors do not explain how the mapping between sentences and Wikipedia titles is done. In addition, their approach is strongly linked to news articles because of the nature of Wikipedia titles. For books like novels that do not have their concepts in Wikipedia, their approach will provide bad summaries. Moreover, their method to compute sentence scores for ranking is not justified and the number of iterations is not defined.

Ledeneva et al. (Ledeneva, García-Hernández, & Gelbukh, 2014) present an extractive text summarization making use of graph-based ranking algorithms. Their proposal consists in detecting Maximal Frequent Sequences as nodes of a graph, and ranking them using a graph-based algorithm such as TextRank or PageRank. In their contribution, these authors do not clearly show how they define a relation between two graph nodes (i.e., terms); they only mention the possibility of using lexical or semantic relations.

Like (Premjith et al., 2015), Mendoza et al. (Mendoza, Bonilla, Noguera, Cobos, & León, 2014) address the generation of extractive summaries from a single document as a binary optimization problem. They define their objective function based on the weighting of individual statistical features of each sentence, such as position, length and the relation between the summary and the title, combined with group features based on the similarity between sentences in each candidate summary and in the original document and between sentences in the summary, in order to obtain coverage of the summary and cohesion of summary sentences. For the optimization, they make use of a memetic algorithm that aims to maximize the objective function for each probable summary. The drawback of their approach is the predefinition of coefficients of the objective function. In addition, the number of iterations to find the best summary is costly.

To sum up, various solutions for ATS are proposed in the literature (CELEBI & DOKUN, 2015; Fang et al., 2015; Hasan & Ng, 2014; Z. He et al., 2015; Ledeneva et al., 2014; Mendoza et al., 2014; Premjith et al., 2015; Sankarasubramaniam et al., 2014); however, several drawbacks can be noted:

1. Some solutions are greedy in processing time due to their optimization functions;
2. Several assumptions are made, such as availability of document topic factors, to validate their approaches;
3. Existing text summarization approaches cannot be applied to scientific papers; they need to be extended and adapted to take into account the specificities of scientific papers in terms of document organization and stock phrases.

In summary, a number of ATS research issues still need to be tackled.

1.3.2.2 Scientific paper summarization

Several models, techniques and algorithms for scientific paper summarization are proposed in the literature, mainly based on MLM and TDM approaches (Dyas-Correia & Alexopoulos, 2014).

Ronzano and Saggion (Ronzano & Saggion, 2016) investigated to what extent citations of a paper are useful to create an improved summary of its content. They analyze how the contents of different parts of a paper, including abstract, body and references, contribute to a widespread summary evaluation metric. In their approach, each citation in a citing paper is manually annotated by four annotators who were asked to identify:

1. The citation context, consisting of one to three text spans in the reference paper and including the related in-line citation marker for the cited paper;
2. The citing spans, consisting of one to three text spans in the other papers which indicate what the reference paper mentioned about the cited paper.

Next, based on TF-IDF applied to the reference paper (first level of citing paper) and citing papers of the reference paper (second level of cited paper), they summarize the cited paper. The main drawback of this approach is that each citation of each citing paper has been manually annotated by four annotators. In addition, their approach is limited to single scientific paper summarization.

Widyantoro and Amin (Widyantoro & Amin, 2014) propose an approach based on citation sentence identification and categorization for generating related-work summaries. Their approach extracts citation sentences and identifies important features for classification of citation sentences that belong to the Problem, Method and Conclusion rhetorical categories. The classification of rhetorical categories uses an MLM approach that requires a training dataset to create a classification model; this classification model is next used as the basis to predict a new sentence rhetorical category. Their classification model is based on the feature set for sentence representation and the specific learning algorithm. They represent a sentence as a feature vector that includes:

1. N-grams,
2. sentence length,
3. thematic word,
4. cue phrase.

For example, the unigram, bi-gram and tri-gram term frequencies are used as features; for each rhetorical category, the authors also use thematic word features selected from sentences in the training set belonging to that category, and the cue phrase feature is a Boolean value that indicates the presence or absence of a cue phrase for the Problem, Method or Conclusion rhetorical category. As in (Ronzano & Saggion, 2016), their approach is limited to single scientific paper summarization. In addition, they do not mention how they obtain the cue phrases for Problem, Method or Conclusion.

Carlos and Thiago (Carlos & Thiago, 2015) present a solution for text mining scientific articles using the R language in the "Knowledge Extraction and Machine Learning" course based on social network analysis, topic models and bipartite graphs. They define a bipartite graph between documents and topics, built with the LDA topic model. In their abstract, these authors claim that they propose a solution for the summarization of abstracts; however, the rest of paper does not explain how the summarization is performed.

Pedram and Omid (Pedram & Omid, 2015) propose a scientific document clustering based on text summarization. Their proposed algorithm consists of four main phases:

1. preprocessing,
2. word weighting and scoring,
3. summarization,
4. clustering.

For the word weighting and scoring phase, TF-IDF is calculated for each word at the document level and okapi BM25 (Best Matching) is calculated at the sentence level. For the summarization phase, the objective of these authors is to remove non-important words; thus, they remove words with a computed BM25 of less than one. Scientific paper summarization cannot be performed in the same way as regular text.

Huang and Wan (Huang & Wan, 2013) propose a novel system, called Academic Knowledge Miner (AKMiner), that mines useful knowledge from articles in a specific domain. Their system extracts academic concepts and relations from academic literature based on a Markov Logic Network. In their approach, these authors focus on two kinds of academic concept: Task and Method. Task concepts are specific problems to be solved in academic literature, while Method concepts are defined as ways to solve specific tasks. They also define two types of relations:

1. Method-Task relations,
2. Method-Method or Task-Task relations.

Method-Task relations refer to the application of a Method to a referred Task, while the second type of relations (between Methods or between Tasks) are formed by dependency, evolution and enhancements. Based on these definitions, the authors make use of Markov Logic Network to extract concepts and relations from academic literature. They apply the first-order knowledge base that is a set of formulae in first-order logic where the predicates and functions are used to describe properties and relations among objects. In their work, all the keywords are collected and summarized manually; they investigated by reading numerous articles and collected four lists of keywords. As in (Ronzano & Saggion, 2016; Widyantoro & Amin, 2014), their approach is limited to single scientific paper summarization.

Caragea et al. (Caragea, Bulgarov, Godea, & Das Gollapalli, 2014) present an approach, called citation enhanced keyphrase extraction (CeKE), that extracts keyphrases from research papers based on information contained in the paper itself and information from the paper's local neighborhood, available in citation networks thanks to the learned models. First, to extract the keyphrases based on TF-IDF, the position of the first occurrence of a phrase is divided by the total number of tokens and the part-of-speech tag of the phrase. Then, they check if the extracted keyphrases occur in cited contexts (paper to summarize is cited by other papers) and citing contexts (paper to summarize is citing other papers) and compute the TF-IDF value of the phrase, computed from the aggregated citation contexts. Citing context is not necessary to summarize a scientific paper; only the text spans in cited context papers related to the paper to summarize are necessary. In addition, their approach requires manual annotation of keyphrases

for training. As in (Huang & Wan, 2013; Ronzano & Saggion, 2016; Widyantoro & Amin, 2014), their approach is limited to single scientific paper summarization.

From this analysis of works about automatic scientific paper summarization (Caragea et al., 2014; Carlos & Thiago, 2015; Huang & Wan, 2013; Pedram & Omid, 2015; Ronzano & Saggion, 2016; Widyantoro & Amin, 2014), it can be observed that:

1. Single scientific paper summarization approaches cannot be used to produce an LR;
2. Some of the approaches need manual contributions;
3. Some works limit the summarization to the identification of keywords or key phrases and ignore the semantic particularities of scientific papers, applying only conventional text summarization techniques.

In the context of this thesis, the focus is on multi-document summarization in order to assist in providing an Assisted Literature Review (ALR).

1.3.3 Automatic multi-document summarization for literature review

For an LR, numerous publications need to be analyzed and summarized; this is referred to as multi-document summarization. In the context of scientific research, given a set of scientific papers, multi-document summarization makes it possible to generate an ALR; however, different styles of LR may be required. According to (Jaidka, Khoo, & Na, 2010), LRs are written in two main styles:

1. A descriptive LR presents critical summaries within a research domain, summarizing individual papers/studies and providing more information about each, such as research methods and results. It focuses on previous studies in terms of approach, results and evaluation. These reviews use sentence templates to perform rhetorical functions;
2. An integrative LR focuses on the ideas and results extracted from a number of research papers and provides fewer details on individual papers/studies.

For researchers with less experience, a descriptive LR with more details about individual studies is more useful. For those who prefer to understand the bigger picture and the main

themes of the research, an integrative LR is better suited. In the present study, the focus is on descriptive ALRs.

Yeloglu et al. (Yeloglu, Milios, & Zincir-Heywood, 2011) investigated four approaches for scientific corpora summarization when only standard key terms are available:

1. original MEAD with built-in default vocabulary,
2. extended MEAD with corpus-specific vocabulary extracted by Keyphrase Extraction Algorithm (KEA),
3. LexRank, a state-of-the-art summarization algorithm based on random walk,
4. W3SS, a summarization algorithm based on keyword density.

Their results show that adding a corpus-specific vocabulary to the MEAD summarization process slightly improves performance; they also determined that LexRank is proven to be impracticable for multi-document summarization of the full texts of scientific documents.

The ALR literature consists of only a few studies. Zajic et al. (Zajic, Dorr, Lin, & Schwartz, 2007) introduce the multi-candidate reduction (MCR) framework for multi-document summarization, in which many compressed candidates are generated for each source sentence; their strategy consists in transitioning from single-document summarization to multi-document summarization. The basic premise of their approach is the construction of a textual summary based on the selection of a subset of words. To do so, they use two algorithms:

1. Trimmer,
2. Hidden Markov Model HEaDline GEnerator (HMM Hedge).

Trimmer selects sub-sequences of words using a linguistically motivated algorithm, while HMM Hedge finds the sub-sequence of words most likely to be a headline for a given story. In other words, sentence selection algorithms are applied to determine which compressed candidates provide the best combination of topic coverage and brevity.

Dunne et al. (Dunne, Shneiderman, Gove, Klavans, & Dorr, 2012) present the results of their effort to integrate statistics, text analytics and visualization in a prototype interface for researchers and analysts. Their prototype system, called Action Science Explorer (ASE),

provides an environment for demonstrating principles of coordination and conducting iterative usability tests of them with interested and knowledgeable users. According to these authors, ASE is designed to support exploration of a collection of papers so as to rapidly provide a summary, while identifying key papers, topics and research groups. ASE uses:

1. bibliometrics lexical link mining to create a citation network for a field and text for each citation,
2. automatic summarization techniques to extract key points from papers using the approach proposed in (Zajic et al., 2007),
3. network analysis and visualization tools to aid in the exploration of relationships.

The first drawback of ASE is that it does not propose an algorithm or model to evaluate the relevancy of a scientific paper in its research field. It uses only bibliometrics for paper ranking. Nor do the authors explain how ASE extracts the sentences containing the citations and their locations from the full text of each paper. In addition, they do not propose a scientific paper summarization approach but simply use the existing algorithm in (Zajic et al., 2007).

Jaidka et al. (Jaidka et al., 2010) present an overview of a project to develop an LR generation system that automatically summarizes a set of research papers using techniques drawn from human summarization behavior. With a view to developing a summarization system that mimics the characteristics of human LR, they try to understand how information is selected from source papers, structured, synthesized and expressed linguistically to support a research study. They analyze and identify:

1. The typical discourse structures and rhetorical devices used in human-generated literature reviews, and the linguistic expressions used to link information in the text to form a cohesive and coherent review;
2. How information is selected from source papers and organized and synthesized in an LR; this aspect is expanded upon in (Jaidka, Khoo, & Na, 2013b).

The authors present only a high-level description of automatic LR. More importantly, they do not propose techniques or algorithms to select relevant scientific papers for a given research

domain or topic. Nevertheless, their study identifies the abstract, conclusion and methodology as the sections of scientific papers used by humans to produce an LR. They also claim that:

1. For a descriptive LR, text from individual sources is copy-pasted or paraphrased;
2. For an integrative LR, inferencing and generalization techniques are used to summarize information from several source papers into a higher-level overview.

J. Chen and Zhuge (J. Chen & Zhuge, 2014) propose a citation-based method for summarizing multiple scientific papers. Their approach is based on the assumption that citation sentences usually talk about a common fact, which is usually represented as a set of noun phrases co-occurring in citation texts and usually discussed from different aspects. Based on this assumption, they designed a multi-document summarization system based on common fact detection. Their main challenge was that citations may not use the same terms to refer to a common fact; to overcome this challenge, they use a term association discovery algorithm to expand terms based on a large set of scientific paper abstracts. Their process is as follows:

1. First, they construct a term co-occurrence base based on the computation of frequently co-occurring terms in the abstracts, titles or even conclusions of a set of scientific papers; they parse the citation sentences to get the noun phrases, from which they generate term bigrams and trigrams and expand the terms based on the term co-occurrence base;
2. Second, they detect common facts in citations and then use them to cluster the citations;
3. Third, they find a subset of the most relevant sentences and form a summary; they treat common facts as a saliency term set where each member term is weighted and is used to score sentences. Based on the Maximal Marginal Relevance (MMR) algorithm, they eliminate redundancy in the sentence set, and to compute the score of each sentence, they make use of a topic signature-based approach. This method first computes a set of terms that relate to a topic and then summarizes documents based on the computed term set.

As in several other works, these authors applied existing algorithms to their architecture.

Agarwal et al. (Agarwal, Gvr, Reddy, & Rose, 2011) present an interactive multi-document summarization system for scientific articles, called SciSumm, that summarizes a set of papers

cited together within the same source article, i.e., co-citation papers. The main idea of the approach is a topic-based clustering of fragments extracted from each cited paper. This analysis enables the generation of an overview of common themes from the co-cited papers. Unfortunately, SciSumm presents some limitations:

1. To obtain the list of relevant articles, SciSumm uses standard retrieval from a Lucene index;
2. The user can use the title, snippet summary and author information to find an article of interest;
3. SciSumm summarizes only the set of cited papers of the citing paper; this summarization task is limited to extracting citation sentences from the citing paper.

Patil and Mahajan (Patil & Mahajan, 2012) present the extension of their previous system for summarizing domain-specific scientific research articles. Based on abstracts and introductions from which any formulae, tables, figures LATEX markups and citations from text files have been removed, they identify the Research Relevant Novelty (RRN) terms—such as goal, method, outcome, contrast & like, continuation—for each category of research. Next, sentences containing the identified RRN terms are extracted and clustered by category. Finally, they use the MMR metric to compute the similarity between multiple sentences. In order to keep only one sentence per cluster of similar sentences, they compute the score of each of them based on the sum of the TF-IDF of the terms of the sentence. As in (Agarwal et al., 2011; J. Chen & Zhuge, 2014; Dunne et al., 2012), these authors make use of existing algorithms.

Jaidka et al. (Jaidka, Khoo, & Na, 2013a) propose an LR framework that contains applications in automatic summarization of scientific papers. This proposal is the extension of their previous contribution (Jaidka et al., 2010). They carry out an analysis of the discourse structure of a sample of 30 literature review sections in research papers in terms of:

1. Macro-level document structure, which makes it possible to identify the different sections of the document, the types of information they contain and their hierarchical organization;
2. Sentence-level rhetorical structure, which reveals how sentences are framed according to the overall purpose of the literature review;

3. Summarization strategies, which show how information was selected and synthesized for the literature review.

For the document structure and rhetorical structure, the authors manually annotate sentences with tags; for example, the topic description tags "Previous research focused on" or "Research in the area of" are used to present a broad overview of research or its context, while the study description tag "In a study by" is used to cite an author and "X identified…", "Y has conducted an experiment to…" are used to describe research processes. The main drawback of their approach is that they do not apply MLM to reduce the manual contributions.

From these related works, it can be seen that the main drawbacks of existing ALR approaches are as follows:

1. Conventional text summarization techniques cannot be applied to scientific research documents; indeed, scientific research documents have a specific structural organization that is different from that of other documents such as narrative or biographical texts. Conventional techniques must be adapted to take into account the specificities of scientific papers in terms of document organization;
2. Most existing approaches are designed for a single document;
3. Certain approaches do not propose new techniques or algorithms, simply making use of existing MLM as well as text and data mining approaches;
4. Even if they propose new algorithms or techniques, they ignore the need to identify scientific papers related to the Researcher Selection in terms of research domain, research specific topic, matching keywords and description of research subject.

The following limitations of existing approaches (Agarwal et al., 2011; J. Chen & Zhuge, 2014; Dunne et al., 2012; Jaidka et al., 2010, 2013a, 2013b; Patil & Mahajan, 2012; Yeloglu et al., 2011; Zajic et al., 2007) should be addressed in the proposed ecosystem:

1. scientific paper ranking,
2. scientific paper summarization,
3. assisted literature review.

CHAPTER 2

MAJOR THEMES

How users search, discover and rank contents and events is of crucial importance, especially with the rapidly increasing volume of data and metadata. This thesis presents the software ecosystem SMESE, which aggregates metadata and data from linked open data, structured data and the metadata authority to create a universal semantic metadata master catalogue using a SPLE model. In this thesis, the advanced versions of the first SMESE prototype are also presented: SMESE V3 and STELLAR V1.

SMESE V1 is the first version of a prototype able to harvest and enrich metadata based on the proposed ecosystem. Its key contributions are:

1. Design and prototyping of a master model that integrates several content types based on a universal metadata model;
2. Definition and prototyping of a mapping ontology in order to allow interoperability between existing metadata models;
3. Definition and prototyping of a software ecosystem architecture that configures an application with software and metadata aspects based on a SPLE model;
4. The proposed SPLE model supports a dynamic metadata CBSD approach creating a harvesting ecosystem for DLs;
5. Prototyping of different processes to increase the findability of related content through interest-based search and discovery engines.

More specifically, the proposed SPLE approach is a combination of feature-oriented reuse method (FORM) and component-oriented platform architecting (COPA) approaches focusing on data and metadata enrichment. With respect to CBSD, SMESE V1 includes a method for selecting composer components for the design of an SPLE. This method can manage and control the complexities of the component selection problem in the creation of the defined product line.

A number of prototypes, experiments and simulations have been conducted to assess the performance of the proposed ecosystem by comparing it against existing enriched metadata techniques or manual LR.

In this thesis, advanced versions of SMESE V1 are also presented: prototype 2 (SMESE V3) and prototype 3 (STELLAR V1). Test results show that SMESE V3 and STELLAR V1 allow greater iterative interpretation of content for purposes of interest-based or emotion-based search and discovery.

SMESE V3, the extended version of SMESE V1, offers the following key contributions:

1. Discovery of enriched sentiment and emotion metadata hidden within the text or linked to multimedia structure using the proposed BM-SSEA algorithm;
2. Generation of semantic topics by text, and multimedia content analysis using the proposed BM-SATD algorithm;
3. Integration of the emotion lexicon of the National Research Council of Canada;
4. Integration and adaptation of a repository of 43 thesauri for semantical contextualization of concepts;
5. Integration of extended LDA and KeyGraph approaches for topic modeling.

STELLAR V1 is a research assistant for the iterative search of relevant papers and production of an Assisted Literature Review (ALR) for a specific subject or topic of research. The key contributions of STELLAR V1 are:

1. The definition of new metadata for scientific content that allow topic-based ranking and relevant paper identification;
2. Classification of metadata in the researcher selection (RS) and researcher annotation (RA) categories;
3. The ability to semantically harvest the web to create a Universal Research Document Repository (URDR) according to RS and from the SMESE V3 ecosystem;
4. The concept of Assisted Literature Review Object (ALRO), which is useful for managing all objects in the ALR. It is basically a component type that includes many types of information useful in producing an ALR;

5. The Literature Corpus Radius (LCR) process, which calculates the distance of each paper to the literature corpus centre for a specific topic, concept or area of research;
6. Machine Learning Models (MLMs), which help researchers to discover, find, rank and refine the iterative list of relevant recommended papers for the creation and enrichment of a final ALR.

This thesis is divided into three sections corresponding to the three technical reports in Appendix I to III:

1. SMESE V1: A Semantic Metadata Enrichment Software Ecosystem is the first prototype;
2. SMESE V3: An ecosystem for topics and emotions that is an extension of the original SMESE V1 is the second prototype;
3. STELLAR V1: An Assisted Literature Review using MLMs to recommend relevant papers and help researchers to build an ALR. STELLAR V1 represents the third prototype and uses the SMESE V3 ecosystem.

2.1 A Semantic Metadata Enrichment Software Ecosystem (SMESE) Based on a Multiplatform Metadata Model for DLs

The first technical report (Appendix I) presents the multiplatform metadata model, an ecosystem for harvesting metadata (including often the data) and internally and externally metadata enrichment for DLs. Metadata are structured information that describes, explains, locates, accesses, retrieves, uses or manages an information resource of any kind. “Metadata” literally means data about data. Some use it to refer to machine understandable information, while others employ it only for records that describe electronic resources. In the library ecosystem, the term is commonly used for any formal scheme of resource description, applying to any type of object, digital or non-digital.

The first prototype of the proposed SMESE V1 architecture is based on SPLE and CBSD approaches to support metadata and entity social and semantic enrichment for DLs. SMESE V1 is based on a mobile first design (MFD) approach for multiplatform user interface. Each

component of the SMESE V1 architecture is based on existing approaches (SPLE and CBSD) and a SME concept (proposed in this work) to generate, extract, discover and enrich metadata. The SME process of SMESE V1 is based on a proposed mapping ontology that makes use of content analysis (internal) and linked data analysis (external).

The main focus of SMESE V1 is metadata meta-modeling, which makes it possible to design different type of content (i.e., metadata content definition) and harvest different source according to their metadata model. For the new generation of information and data management, metadata are a highly efficient material for data aggregation. For example, it is easier to find a specific set of user interests when metadata such as content topics or sentiments are available in the enriched model. Furthermore, it is possible to increase user satisfaction by reducing the user interest gap. To make this feasible, all content needs to be enriched. In other words, specific metadata must be available including semantic topics, sentiments and abstracts. However, at the present time, most content does not have these metadata.

The SMESE V1 multiplatform prototype aggregates multiple world catalogues from libraries, universities, bookstores, #tag collections, museums, open catalogues, national catalogues and others. It harvests and processes metadata from full-text content (where possible).

Central indexes typically include full text and citations from publishers, full text and metadata from open-source collections, full text, abstracting and indexing from aggregators and subscription databases, and different formats (such as MARC) from library catalogues, also called the base index, unified index, or foundation index.

The SMESE V1 multiplatform framework try to link bibliographic records and semantic metadata enrichments (SEM) into a master metadata catalogue. This catalogue includes collections or novelties as: papers, books, DVDs, CDs, comics, games, pictures, videos, legacy collections, organizations, rewards, TV, radio, and museums.

Figure 2.1 presents the four levels of the semantic collaborative gateway in SMESE V1:

1. Meta-Entity (*black*),
2. Entity (*blue*),

3. Semantic metadata enrichment and creation (*grey*),
4. Contents & Events (*white*).

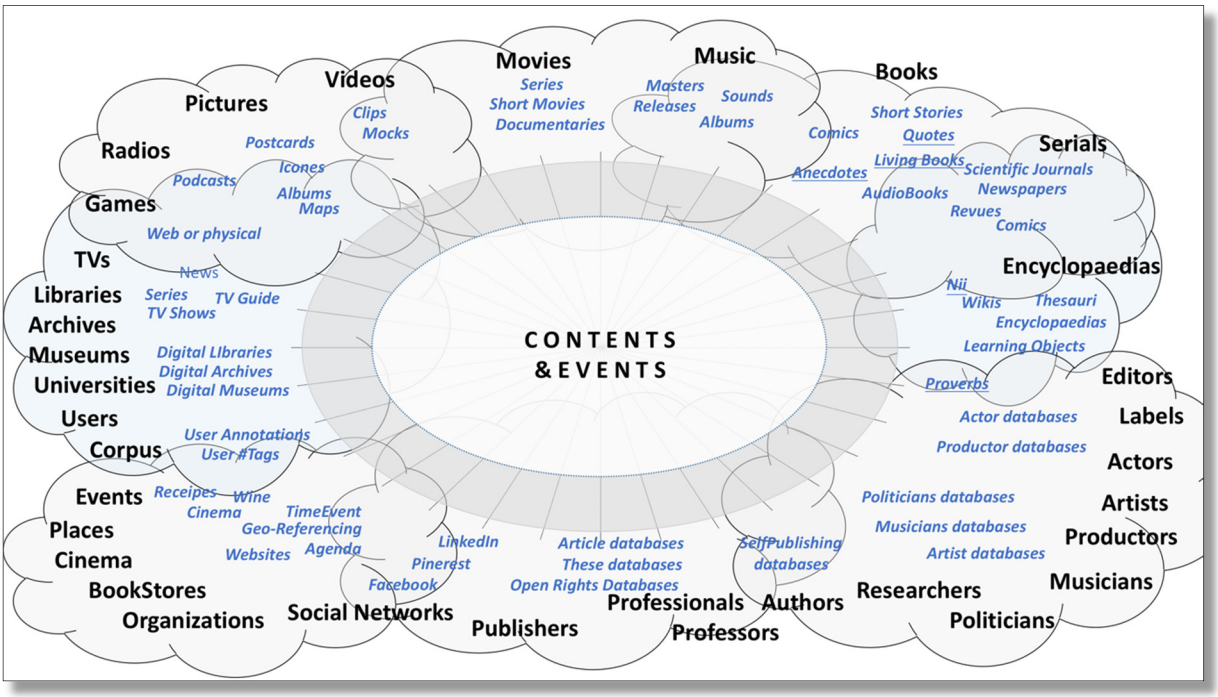


Figure 2.1 Meta-model and metadata enrichment view

Semantic relationships between content, persons, organizations, events and places are defined and curated in the master metadata catalogue. Topics, sentiments and emotions are extracted (where possible) from the content, its context and related objects. As semantic relationships between the content and users who are persons, the new metadata (interests, topics and emotions) are defined and may be extracted (where possible) from the content, its context and related objects.

SMESE V1 allows users to find topically related content through an interest-based search and discovery engine. Transforming bibliographic records into semantic data is a complex problem that includes interpreting and enriching the information. Fortunately, many international organizations (e.g., Bibliothèque Nationale de France (BNF), Library of Congress and some others) have done some of this heavy work and already have much bibliographic metadata converted into triple-stores according to defined schemas.

Recent catalogues support the ability to publish and search collections of descriptive entities (described by a list of generic metadata) for data, content and related information objects. Metadata in catalogues represent resource characteristics that can be indexed, queried and displayed by both humans and machine. Catalogue metadata are needed to support the discovery and notification of information within an information community. Using information from specific SME interests and emotions, the ecosystem is able to provide the final user with better results that match his or her interest, emotion or mood.

This new SMESE V1 semantic ecosystem harvest and enrich bibliographic records externally (*from the web or databases*) and internally (*from text data or object*). As shown in Figure 2.2, the main components of the SMESE V1 ecosystem are:

1. metadata initiatives & concordance rules,
2. harvesting of web metadata & data,
3. harvesting of authority metadata & data,
4. rule-based semantic metadata external enrichment,
5. rule-based semantic metadata internal enrichment,
6. semantic metadata external & internal enrichment synchronization,
7. user interest-based gateway,
8. semantic master catalogue,
9. semantic analytical engine.

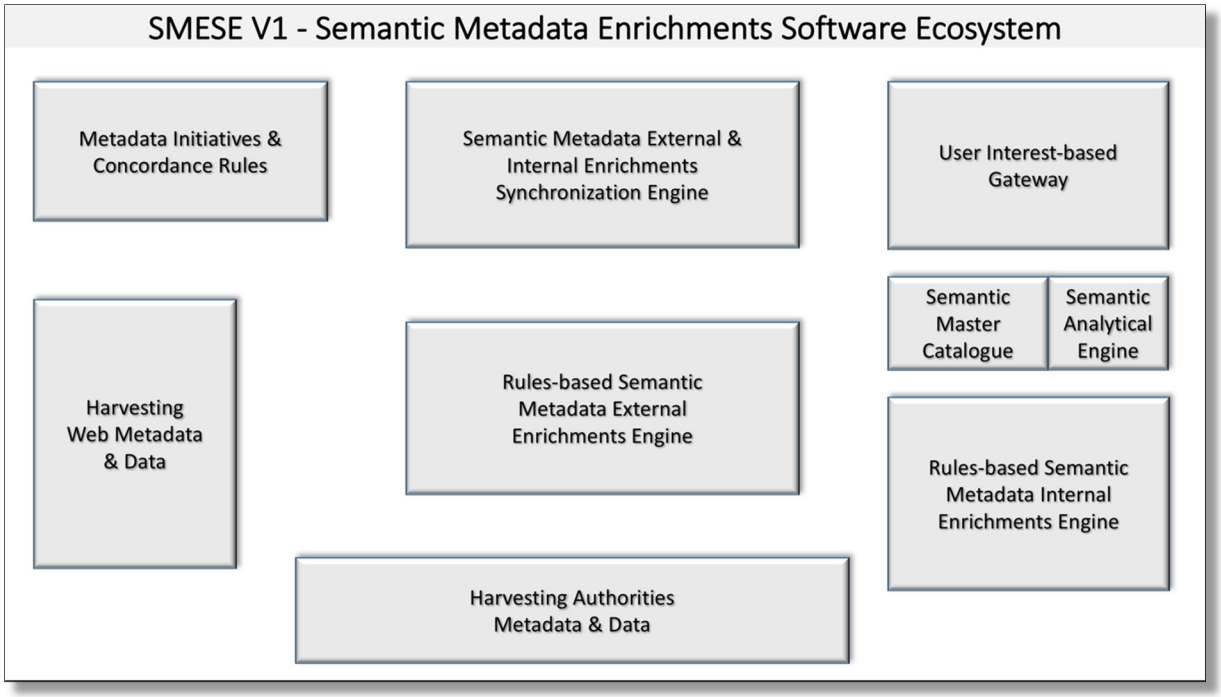


Figure 2.2 Semantic Enriched Metadata Software Ecosystem (SMESE V1) – 1st prototype

Many metadata schemas exist to describe various types of textual and non-textual objects including published books, electronic documents, archival documents, art objects, educational and training materials, scientific datasets and, obviously, the web. Large national and international DL projects, such as Europeana and the Digital Public Library of America, have highlighted the importance of sharing metadata across silos.

Many aggregators harvest metadata that, in the process, may become inaccurate because they did not look at the semantic context. In practice, aggregators usually ignore the idiosyncratic use of metadata schemas and enforce the use of designated metadata fields. Connecting data across silos would help to improve the ability of users to browse and discover related entities (metadata) without having to do multiple searches in multiple portals. The proposed SMESE V1 ecosystem defines crosswalks that create metadata pathways to different sources; each pathway checks the structure of the metadata source and then performs data harvesting. Figure 2.3 shows the semantic metadata meta-catalogue classification designed and implemented in the SMESE V1 prototype.

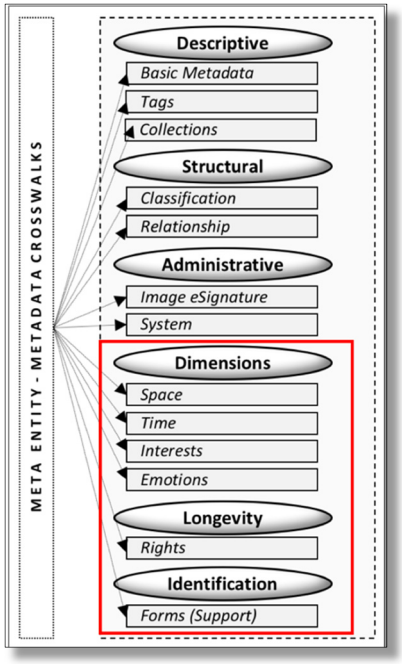


Figure 2.3 Semantic metadata meta-catalogue classification in the SMESE V1 first prototype

Semantic searches over documents and other content need to use semantic metadata enrichment (SME) to find information based not just on the presence of words, but also on their meaning. Linked open data (LOD) based semantic annotation methods are good candidates to enrich the content with disambiguated domain terms and entities (e.g. events, emotions, interests, locations, organizations, persons), described through Unique Resource Identifiers (URIs) (Bontcheva et al., 2015). In addition, the International Standard Names Identifier (ISNI) has been proposed by national libraries to organize and catalogue semantic metadata relationships, see Figure 2.4, adapted from *ISNI, For a Worldwide Identification Ecosystem* (INHA – Institut National de l'histoire de l'art, 11 January 2016, Anila Angjeli, Bibliothèque nationale de France, ISNI 0000 0004 2755 4724). The symbol with three blue dots (RDF) represents a semantic repository using triple stores. The BNF is identifying workflows with publishers to provide them with ISNIs for new authors. The ISNI system is an opportunity to help enrich author metadata and the quality of the authority files. ISNI semantic relationships make it possible to connect many sources of information, including:

1. BNF Catalog,
2. Data.bnf.fr,

3. VIAF,
4. IdRef,
5. Union List of Artist Names,
6. SNAC,
7. AGORHA,
8. Wikidata,
9. Wikipedia,
10. Data.banq.ca (in 2017 for Québec metadata).

Figure 2.4 also shows the introduction of ISNI semantic relationships into the semantic metadata meta-catalogue of the SMESE V1 prototype.

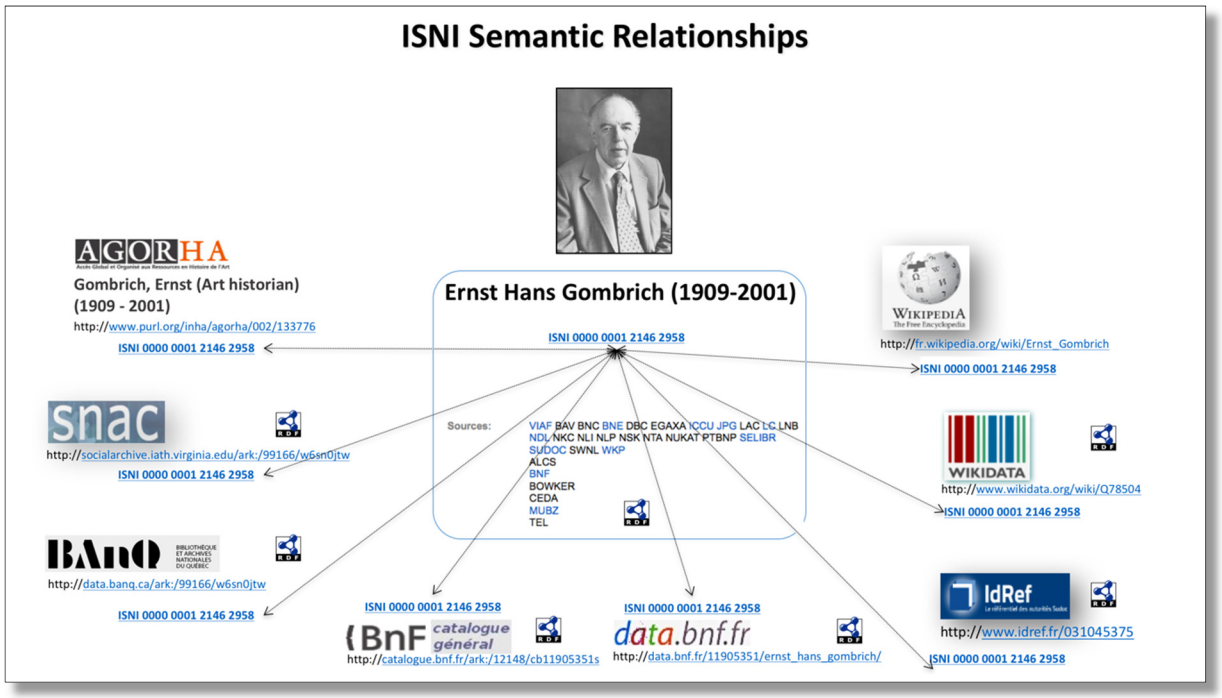


Figure 2.4 ISNI semantic relationships of metadata in the SMESE V1 prototype

The original content should be enriched with relevant knowledge from the respective LOD resources (e.g. that Justin Trudeau is a Canadian politician). This is needed to answer queries that require common-sense knowledge, which is often not present in the original content. For example: following semantic enrichment, a semantic search for events that provide specific

emotions (e.g., happiness, joy, etc.) in Montreal according to individual interests this weekend would provide relevant metadata about events in Montreal, even though not explicitly mentioned in the original content metadata.

The semantic annotation process of SMESE V1 creates relationships between semantic models, such as ontologies and persons. It may be characterized as the semantic enrichment of unstructured and semi-structured content with new knowledge and linking these to relevant domain ontologies and knowledge bases. This requires the use of ISNI, other authority files or other techniques. It typically requires annotating a potentially ambiguous entity mention (e.g. Justin Trudeau) with the canonical identifier of the correct unique entity (e.g. depending on the content, http://dbpedia.org/page/Justin_Trudeau). The benefit of social semantic enrichment is that by surfacing annotated terms derived from the full-text content, concepts buried within the body of the paper or report can be highlighted. The addition of terms also affects the relevance ranking in full-text searches. Moreover, users can be more specific by limiting the search criteria to the subject, interest or emotion metadata (e.g. through faceted search).

These processes extract, analyze and catalogue metadata for topics and emotions involved in the SMESE ecosystem. As today, an amount of 5 millions content have been harvested over a target amount of close to 500 millions, see the Table 2.1 for an overview of the detail about harvested metadata and data (p.e. papers and events) in the prototype. For each content type many metadata and data have been extracted and enriched. These enrichment processes are based on information retrieval and knowledge extraction approaches. The text is analyzed by means of extensions of text mining algorithms such as latent Dirichlet allocation (LDA), latent semantic analysis (LSA), support vector machine (SVM) and k-Means.

Table 2.1 Harvesting statistic related to metadata and data – SMESE V1

Priority	Sources title	Sources url	Status	%	Total Content	Total harvested
1	Book Depository (type = html)	https://www.bookdepository.com/	harvesting	0,9%	26 750 719	231 453
2	opendoar (type = OAI-pmh)	http://www.opendoar.org/	harvesting	0,3%	235 828 824	612 545
3	ResearchGate (type = html)	https://www.researchgate.net/	harvesting	0,4%	110 210 000	453 129
4	Academia (type = html)	https://www.academia.edu/	harvesting	5,0%	15 363 735	765 345
5	Amazon (type = html)	http://www.amazon.com/	harvesting	9,7%	4 703 063	456 234
6	Paulines (type = html)	http://www.paulines.qc.ca/	harvesting	98,1%	171 120	167 890
7	L'exèdre (type = html)	http://exedre.leslibrairies.ca/	harvesting	78,4%	171 120	134 120
8	FNAC (type = html)	http://www.fnac.com/	harvesting	66,6%	156 224	104 109
9	Poirier (type = html)	http://www.librairiepoirier.ca/	Novelties	100,0%	176 162	176 162
10	Archambault (type = html)	http://www.archambault.ca/	Novelties	100,0%	165 405	165 405
11	Renaud-Bray (type = html)	http://www.renaudbray.com/accueil.aspx	Novelties	100,0%	347 380	347 380
12	COOP UdeS (type = html)	http://www.usherbrooke.coop/	Novelties	100,0%	47 412	47 412
13	Biblairie GGC (type = html)	http://www.biblairie.qc.ca/	Novelties	100,0%	213	213
14	Librairie Media Paul (type = html)	http://librairiemediaspaul.ca	Novelties	100,0%	29 938	29 938
15	Leslibraires.fr (type = html)	https://www.leslibraires.fr/	finished	100,0%	888 750	888 750
16	Mollat (type = html)	http://www.mollat.com/	Novelties	100,0%	505 729	505 729
17	academic microsoft (type = html)	https://academic.microsoft.com/	harvesting	0,6%	80 000 000	453 240
18	Coop HEC (type = html)	http://www.coophec.com/	harvesting	7,3%	17 222	1 265
19	thinkmind (type = html)	http://www.thinkmind.org/	to be started	0,0%	56 744	-
22	PBS	http://www.pbs.org/	harvesting	5,1%	5 434	279
23	World	http://worldchannel.org/	to be started	0,0%	34 211	-
24	City	http://www.citytv.com/montreal/	harvesting	45,3%	12 110	5 489
25	WCAX (Local News and Weather)	http://www.wcax.com	to be started	0,0%	7 549	-
26	TVHebdo	http://www.tvhebdo.com	harvesting	56,0%	14 231	7 972
31	justWatch	http://www.justwatch.com	harvesting	25,6%	54 345	13 927
			TOTAL:		475 717 640	5 567 986

One of the contributions of SMESE V1 for DLs is that it is not specific to one software product but can be applied to many products dynamically. In addition, it includes a semantic metadata enrichment (SME) process to improve the quality of search and discovery engines.

Note that metadata modeling and an universal metadata model is the main focus of SMESE V1. The proposed SECO of SMESE V1 uses an SPLE architecture that is a combination of FORM and COPA to catalogue semantically different contents.

The SECO of SMESE V1 also proposes a decision support process called SPLE-DSP. SPLE-DSP supports the activation and deactivation of software features related to metadata and takes into account automatic runtime reconfiguration according to different scenarios. In addition, SPLE-DSP rebinds to new services dynamically based on the description of the relationships and transitions between multiple binding times under an SPLE when the software adapts its system properties to a new context. To take context variability into account in modeling

context-aware properties, SPLE-DSP makes use of an autonomous process that exploits context information to adapt software behavior using a universal metadata model.

Furthermore, SPLE-DSP integrates the adaptation of metadata and products dynamically. This helps products to evolve autonomously when the environment changes and provides self-adaptive and optimized reconfiguration.

This reconfiguration model, called dynamic and optimized metadata-based reconfiguration model (DOMRM), takes into account the preferences of several users who have different requirements in terms of desirable features and measurable criteria.

When the user chooses preferences in terms of system behavior, the semantic weight of each feature is computed based on the software feature configuration model (FCM). FCM represents the semantic relationship between features where each feature is active or not. In addition, FCM defines the rules that control the activation status of each feature according to its links with other features. For example, a rule may be: feature Fi should never be activated when Fi-1 is activated. Based on this rule, the FCM automatically activates or deactivates the feature.

The rules are also used to predict the behavior of the application based on the activation status of features according to users' selections. Note that individual users have their own weight per feature, defined on the basis of that user's use of the feature. This weight quantifies the importance of the feature for the user.

2.2 A Semantic Metadata Enrichment Software Ecosystem Based on Sentiment and Emotion Analysis Enrichment (SMESE V3)

The second technical report (Appendix II) focuses on contributions designed and implemented in the SMESE V3 prototype in two research fields: semantic topic detection (STD) and sentiment analysis (SA).

2.2.1 Semantic topic detection

Semantic topic detection (STD), a fundamental aspect of SIR, helps users efficiently detect meaningful topics. It has attracted significant research in several communities in the last decade, including public opinion monitoring, decision support, emergency management and social media modeling (Hurtado et al., 2016; Sayyadi & Raschid, 2013). STD is based on large and noisy data collections such as social media, and addresses both scalability and accuracy challenges. Initial methods for STD relied on clustering documents based on a core group of keywords representing a specific topic, where, based on a ratio such as TF-IDF, documents that contain these keywords are similar to each other (Niu et al., 2016; Salton & Buckley, 1988). Next, variations of TF-IDF were used to compute keyword-based feature values, and cosine similarity was used as a similarity (or distance) measure to cluster documents. The following generation of STD approaches, including those based on latent Dirichlet allocation (LDA), shifted analysis from directly clustering documents to clustering keywords. Some examples of these advances in STD are presented in (David M. Blei et al., 2003).

However, social media collections differ along several lines, including the size distribution of documents and the distribution of words. One research challenge is to rapidly filter out noisy and irrelevant documents, while at the same time accurately clustering a large collection. Bijalwan et al. (Bijalwan et al., 2014), for example, experimented with machine learning approaches for text and document mining and concluded that k-nearest neighbors (KNN), for their data sets, showed the maximum accuracy as compared to naive Bayes and term-graph. The drawback of KNN is that time complexity (i.e., amount of time taken to run) is high but it demonstrates better accuracy than others.

2.2.2 Sentiment analysis (SA)

The main objective of sentiment analysis (SA) is to establish the attitude of a given person with regard to sentences, paragraphs, chapters or documents (Appel et al., 2016; Balazs & Velásquez, 2016; Fernández-Gavilanes, Álvarez-López, Juncal-Martínez, Costa-Montenegro,

& Javier González-Castaño, 2016; Niu et al., 2016; Patel & Madia, 2016; Ravi & Ravi, 2015; Serrano-Guerrero, Olivas, Romero, & Herrera-Viedma, 2015). Many websites offer reviews of items like books, cars, mobile devices, movies etc., where products are described in some detail and rated as good/bad, liked/disliked. With the rapid spread of social media, it has become necessary to categorize these reviews in an automated way (Niu et al., 2016).

There are different ways to perform SA, such as keyword spotting, lexical affinity and statistical methods. However, the most commonly applied techniques belong either to the category of text classification supervised machine learning (SML), which uses methods like naive Bayes, maximum entropy or support vector machine (SVM), or to the category of text classification unsupervised machine learning (UML).

One current limitation in the area of SA research is its focus on sentiment classification while ignoring the detection of emotions. For example, document emotion analysis may help to determine an emotional barometer and give the reader a clear indication of excitement, fear, anxiety, irritability, depression, anger and other such emotions. For this reason, we focus on sentiment and emotion analysis (SEA) instead of SA.

2.2.3 SMESE V3 approach to STD and SEA

Our research has looked to improve the accuracy of topic detection and sentiment and emotion discovery by semantically enriching the metadata from linked open data and the bibliographic records existing in different formats. The second technical report presents the design, implementation and evaluation of the SMESE V3 ecosystem. More specifically, SMESE V3 consists of prototypes implementing two rule-based algorithms to enrich metadata semantically:

1. BM-SATD: generation of semantic topics by text analysis, relationships and multimedia content,
2. BM-SSEA: discovery of sentiments and emotions hidden within the text or linked to a multimedia structure through an Artificial Intelligence (AI) computational approach.

Using simulation, the performance of SMESE V3 was evaluated in terms of accuracy of topic detection and sentiment and emotion discovery. Existing approaches to enriching metadata (e.g., topic detection or sentiment and emotion discovery) were used for comparison. Simulation results showed that the enhanced SMESE outperforms existing approaches.

In Figure 2.5, improvements to the SMESE V3 platform (2nd prototype) stemming from this research work and its implementation are presented in blue.

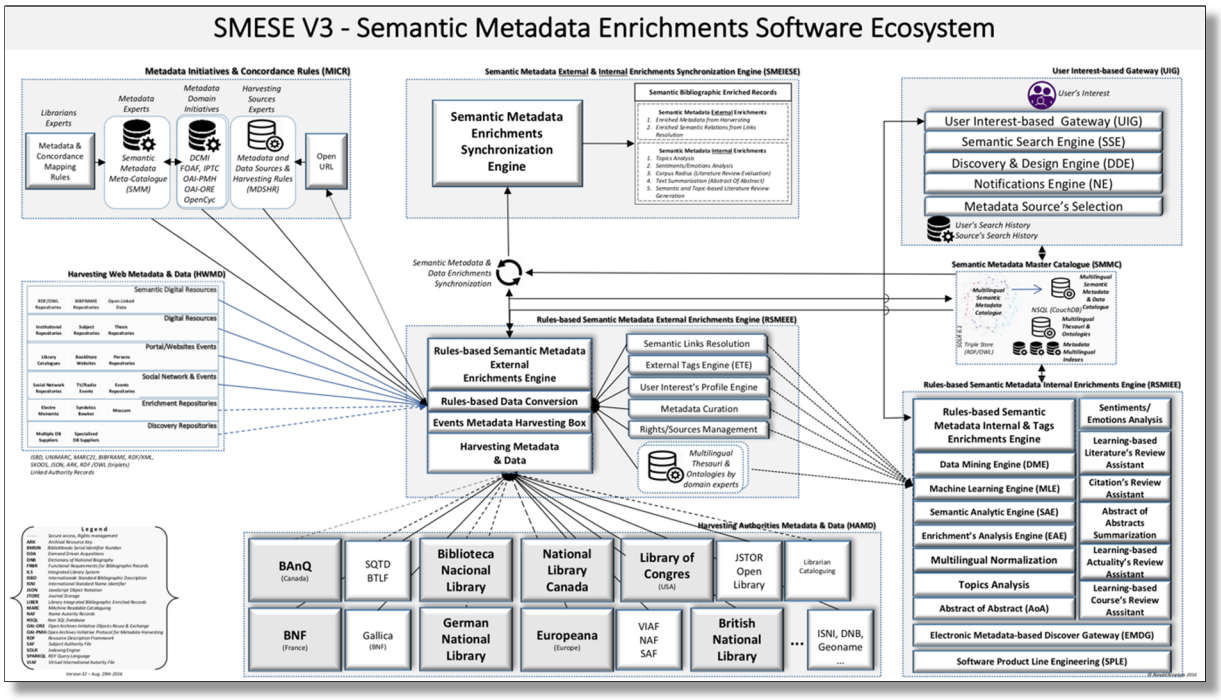


Figure 2.5 SMESE V3 – Semantic Metadata Enrichment Software Ecosystem– 2nd prototype

For more understanding about SMESE V3 algorithms and processes to semantically enrich metadata, refer to Appendix II, which describes in detail this second prototype of SMESE.

2.3 An Assisted Literature Review using Machine Learning Models to Build a Literature Corpus and to Recommend References using their Related Radius from this Corpus

The third technical report (Appendix III) presents another enhanced SMESE prototype that implements an Assisted Literature Review (ALR) design using Machine Learning Models (MLM) to build a literature corpus and to recommend references using their related radius from this corpus. This prototype, called STELLAR V1 (Semantic Topics Ecosystem Learning-based Literature Assisted Review), is more useful for electronic papers (ePapers).

Electronic papers play a critical role in the dissemination of research results through conferences and journals or new channels such as social media. With the evolving and interdisciplinary nature of research, there is an increasing need to develop MLMs that can facilitate and assist researchers in the iterative creation of their LR (i.e., manual literature review). The goal of this third technical report is to define and prototype the automation of a process to assist students, teachers, librarians and other users in producing and maintaining an ALR.

Researchers now acknowledge that ePapers are not sufficient to communicate and share information about research investigations. The volume of scientific publications available is becoming an issue for researchers (Mayr, Scharnhorst, Larsen, Schaer, & Mutschke, 2014). Given that so many literature reviews are incomplete, the lack of automation algorithms to assist in ALR creation and ongoing process is surprising.

A literature review needs to be systematic and focused on user selections, incorporating only things that are relevant to the research topic. It has to be evaluative, assessing each citation to determine its ranking and if it is worth including in the ALR. One of the research goals of the STELLAR V1 prototype is to reduce reading load by helping researchers to read only an intelligent selection of documents. Using TDM, MLMs and a classification model that learns from paper's metadata and user-annotated data, it detects metadata and identifies relevant papers for a literature review in a specific research field and on a specific topic.

Figure 2.6 presents a simplified view of the proposed STELLAR V1 model. Specifically, it shows the MLM processes associated with each step of STELLAR V1 (i.e., those above each step of the ALR).

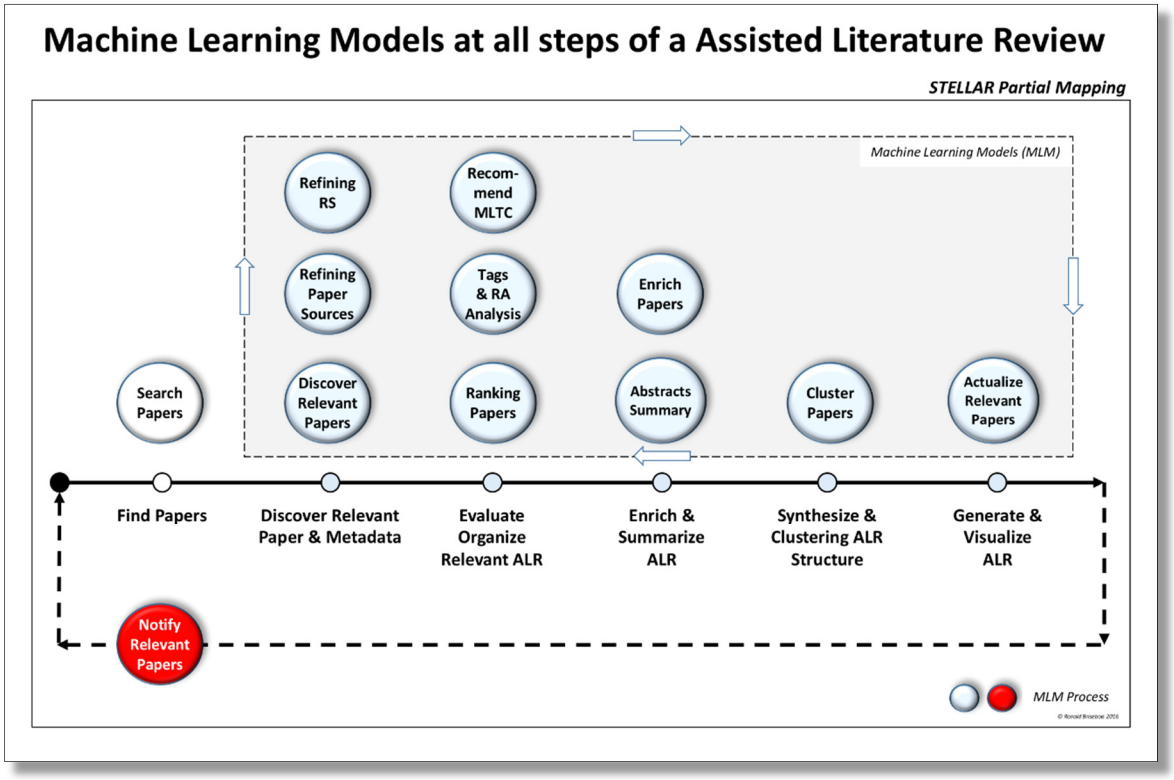


Figure 2.6 MLMs at all steps of an Assisted Literature Review

It takes many steps to produce and deliver a quality LR manually. In the automation of this process, many tools and algorithms have been developed to assist and alert the researcher. Harvesting tools, search engines and MLMs have been used to execute many of the tasks in this process. Figure 2.6 shows the iterative process of creating an ALR using MLMs. This process helps the researcher to find, rank and tag the relevant papers, and to receive recommendations about how to improve the literature review on an ongoing basis. It also notifies the researcher when a new paper concerning his or her research topic is published or available. The MLMs could be used to learn and improve the process in two ways:

1. For each step in the light blue processes, the MLM are used to refine the results (in Figure 2.6, there are 10 blue circles related to MLMs);

2. The entire process is iterative, so it could be enhanced by discovering dynamically a new relevant paper and notifying the user.

The first step (i.e., Find Papers) does not require an MLM, but the next five do (from *Discover relevant papers and metadata* to *Generate and visualise ALR*). In the same figure, the blue circles represent MLM processes while the white and red circles represent a non-MLM process.

One of the interesting and innovative aspects of this process is to be able to notify the researcher about new papers that meet the RS (Researcher Selection), which is made up of the different metadata describing the research topic or area. This process helps the researcher update the ALR after many months of work on a topic without doing intensive searching as would be required in a manual LR.

The detail view of the proposed STELLAR V1 model is presented in Figure 2.7.

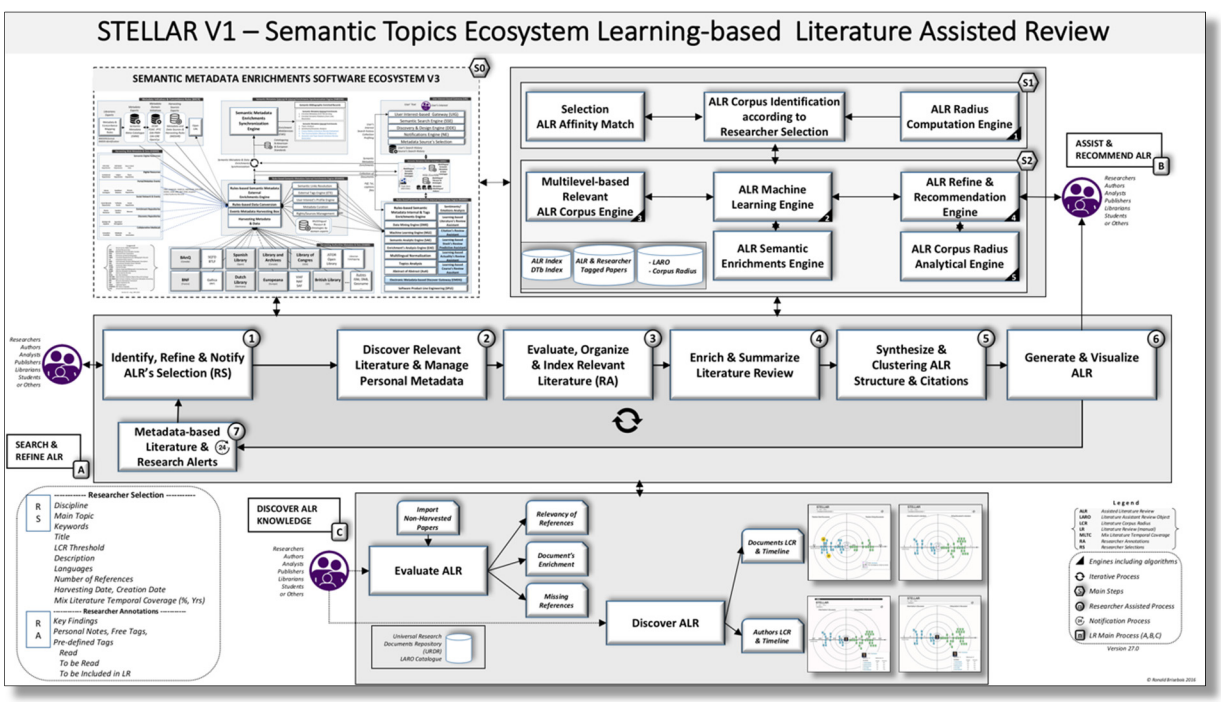


Figure 2.7 STELLAR V1 – Semantic Topics Ecosystem Learning-based Literature Assisted Review – 3rd prototype

There are four main processes designed for STELLAR V1:

1. Search & Refine ALR,
2. Improve ALR by TDM & MLM,
3. Discover ALR,
4. Semantic Metadata Enrichments Software Ecosystem V3.

And there is one outside process named Semantic Metadata Enrichments Software Ecosystem. This process refers to the two other articles defining the SMESE platform and some enrichments (Appendix I for SMESE V1 and II for SMESE V3). The proposed model is an iterative process where the user could Search & Refine the research topic or area by modifying the ALR selections. STELLAR V1 could be used by different types of users such as researchers, authors, publishers, students and librarians.

One of the important aspects of STELLAR V1 is semantic metadata enrichment and ranking of papers. This function draws information from a paper in order to enrich its metadata. In our previous work (Brisebois, Abran, & Nadembega, 2016), two types of semantic enrichment were defined: internal and external. Semantic internal enrichment extracts citations from the document body and automatically produces the abstract (see Figure 2.8).

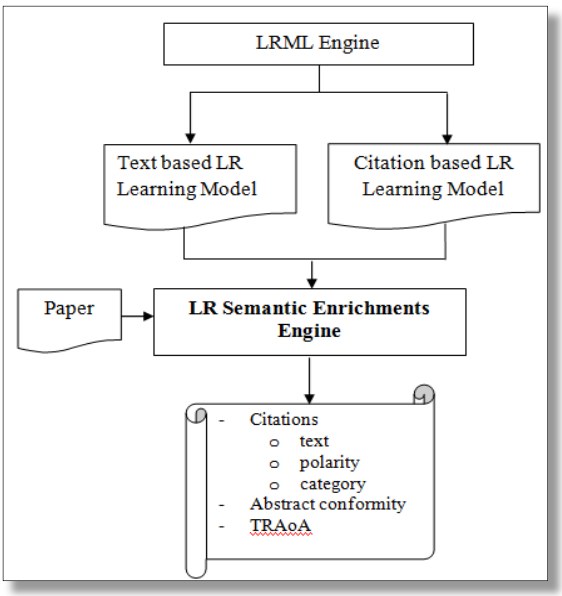


Figure 2.8 STELLAR V1 semantic enrichments TDM

More specifically, the ALR-based MLM provides two types of learning model:

1. A text-based model that may be applied to text according to its section in the document to extract relevant information;
2. A citation-based model that focuses on the context of a citation to extract the citation itself, its polarity (positive or negative) and its category.

Thus, two types of enrichment are considered:

1. citation-based enrichments,
2. abstract conformity-based enrichments.

2.3.1 Citation-based enrichments

The citation-based enrichments learning step identifies the citation sentences (e.g., sentences that contain a citation) and enriches them through a classification process identifying their category and polarity. Each sentence is extracted and analyzed using the citation-based learning model to identify citations in a paper. When a citation is identified, the citation polarity learning model is used to determine its polarity while the citation category learning model is used to categorize the citation.

2.3.2 Abstract conformity-based enrichments

In the STELLAR V1 prototype, the abstract conformity-based enrichment sub-step evaluates the similarity between the abstract and the rest of the document. The conformity evaluation allows a researcher to decide whether or not to read the rest of the document after reading the abstract. It may happen that the abstract claims a solution, new algorithm, new approach or best results not substantiated in the rest of document.

To perform an abstract conformity evaluation, the text-based ALR learning model consists of:

1. A cue phrase learning model that contains a list of cue phrases (CP); CP is used to identify and enrich the text category;

2. A thematic learning model that contains a list of rhetorical expressions of thematics (TR); TR is used to classify the text category.

More specifically, the sub-step identifies, from the abstract and the rest of the document, the set of texts per category. For example: considering the abstract, a set of texts (i.e., category) is identified for Problem, Solution and Result. Next, the text category conformity is evaluated for each category based on the extracted thematic terms using the category rhetorical expression (i.e., P_TR, S_TR and R_TR) of the thematic learning model.

2.3.3 Abstract of Abstracts (AoA) enrichments

In the STELLAR V1 prototype, the enrichment step of the abstract of abstracts (AoA) presents the research topic's evolution over time; here, the term "radius" is used to indicate that all time intervals are represented as a distance between two years, one of which is designated as the center of a circle. The radius expresses the relevancy of a paper according to the researcher selection. Taking the relevant documents published within the same years, their abstracts are extracted and summarized to provide an AoA. For a document, the AoA generation process is similar to the abstract conformity-based enrichment step, but it focuses on the abstract instead of the rest of the document. To produce an AoA, the text-based LR learning model is used. More specifically, the enrichment process identifies a set of abstracts per category and extracts, for each category, the thematic sentences using the category rhetorical expression (e.g., P_TR, S_TR and R_TR) of the thematic learning model. Thus, to obtain the AoA, the corpus of papers is:

1. classified by its temporal radius,
2. applied to each document of each class.

These steps produce an AoA for the corpus of documents. Numerous simulations have been conducted to assess the performance of the prototypes and the results are presented in third technical report (see details in Appendix III).

CONCLUSION

This section presents a summary of the contributions, prototypes and results of this thesis.

The three technical reports that make up the core of these research contributions, and that have been submitted to journals for peer review, are focused on the following research issues:

1. data and metadata semantic harvesting ecosystem using a mapping ontology model for enhance DL's capability,
2. semantic metadata enrichments (SME) based on machine learning models (MLMs) especially for topics and emotions,
3. assisted literature reviews based on MLMs to assist and alert the researcher in producing a literature review.

It was observed that DL users do not have all the semantic metadata needed to make decisions when searching or looking to discover specific contents or a particular event. It is very challenging to:

1. Take advantage of the power of the semantic web, due to the poor quality of metadata in many library collections (i.e., content);
2. Share, merge or search existing content or collections, due to the lack of a unified model for interoperability of metadata models such as Dublin Core, UNIMARC, MARC21, RDF/RDA and BIBFRAME;
3. Identify relevant content, due to the lack of enriched metadata that is easy to understand;
4. Manually enrich metadata, due to the exponential growth of content, the volume of metadata and the number of semantic relationships between content and metadata.

To overcome these challenging issues, which limit the full utilization of content or event, this thesis has proposed a number of contributions that can be employed by users in metadata and data management to better catalogue and enrich content and event. This will allow users to make better decisions in the selection of content or event. For example, researchers will find it easier to identify and prioritize relevant scientific papers for their ALR.

The first technical report focuses on the definition of an interoperable metadata and meta-entity model, called semantic metadata enrichment software ecosystem (SMESE V1), to support digital multiplatform metadata harvesting applications, and more specifically DLs. It also proposes a software product line engineering process that uses a component-based software development approach for integrating content management with multi-applications catalogue. To take into account the interoperability of existing metadata models, SMESE V1 implements an ontology mapping model. SMESE V1 also includes an SPLE decision support process (SPLE-DSP), which is used to support dynamic metadata reconfiguration (see Appendix I).

The main contributions of this first technical report are as follows:

1. Definition of a software ecosystem model that configures the application production process including software aspects based on a proposed CBSD and metadata-based SPLE approach;
2. Definition and partial implementation of semantic metadata enrichment using SPLE and a semantic master metadata catalogue;
3. Definition and prototype of a SECO-based DL standard and interoperable metadata model able to:
    a. take into account interoperability mechanisms to guide the self-adaptation of product compositions according to changes in the client configuration,
    b. take into account several semantic enrichment aspects,
    c. include several enriched metadata and entity models.
4. Design and implementation of a SMESE V1 prototype for a semantic digital library.

The second and third technical reports extend the contributions of the first technical report by focusing on the research field of automatic entity metadata enrichments: semantic topic detection, sentiment and emotion analysis and metadata usage for literature-assisted review objects.

Note that the prototype presented in the second technical report is called SMESE V3. More specifically, this second technical report contains four distinct new contributions:

1. Adaptation of conventional text summarization approaches to take into account the specificities of scientific papers in terms of document organization;

2. Discovery of enriched sentiment and emotion metadata hidden within the text or linked to multimedia structure using the proposed BM-SSEA (BM-Semantic Sentiment and Emotion Analysis) algorithm;
3. Implementation of rule-based semantic metadata internal enrichment (that includes algorithms BM-SATD (BM-Scalable Annotation-based Topic Detection) and BM-SSEA);
4. Generation of semantic topics by text, and multimedia content analysis using the proposed BM-SATD algorithm.

The main research objective in this second technical report was to enhance the SMESE V1 platform through text analysis approaches for topic, sentiment, emotion, and semantic relationship detection. More specifically, BM-SATD fuses multiple relations into a term graph and detects topics from the graph using a graph analytical method (see Appendix II for details). BM-SATD presents a hybrid relation analysis and machine learning approach that integrates semantic relations, semantic annotations and co-occurrence relations for topic detection; it combines semantic relations between terms and co-occurrence relations across the document making use of document annotation. BM-SATD not only detects topics more effectively by combing mutually complementary relations, but also mines important rare topics by leveraging latent co-occurrence relations.

BM-SATD includes:

1. A probabilistic topic detection approach that is an extension of LDA, called BM semantic topic model (BM-SemTopic);
2. A clustering approach that is an extension of KeyGraph, called BM semantic graph (BM-SemGraph).

BM-SSEA classifies the documents taking emotion into consideration; it determines which sentiment a document more likely belongs to (see more details about BM-SSEA in Appendix II). It is a hybrid approach that combines keyword-based and rule-based approaches. In order to take into account the semantic aspect of sentiment and emotion analysis, BM-SSEA uses several semantic lexical resources that create its knowledge. The evaluation of this TDM shows that BM-SATD provides an average accuracy of 79.50% per topic and BM-SSEA

demonstrates an average accuracy of 93.30% per emotion; the details of the simulation results can be seen in Appendix II.

The third technical report proposes an Assisted Literature Review (ALR) prototype, STELLAR V1 (Semantic Topics Ecosystem Learning-based Literature Assisted Review), based on machine learning models and a semantic metadata enrichment ecosystem. It discovers, finds and recommends relevant papers for a literature review in a specific field of research. Using TDM, MLMs and a classification model that learns from researchers' annotated data and semantic enriched metadata, STELLAR V1 identifies, ranks and recommends relevant papers according to the researcher selection, see Figure 2.9.

In this figure, there is a conceptual representation of STELLAR V1. All the rectangles (in any color) represent papers available in a specific domain of knowledge (URDR). The black rectangle are irrelevant papers according to the researcher selection; the one in blue are relevant to the ALR; the one in yellow are part of the suggested selection outside the literature corpus radius (LCR is inside the white circle); the one in red are the researcher annotated papers, who could be inside the ALR Papers Corpus or inside the Literature Corpus.

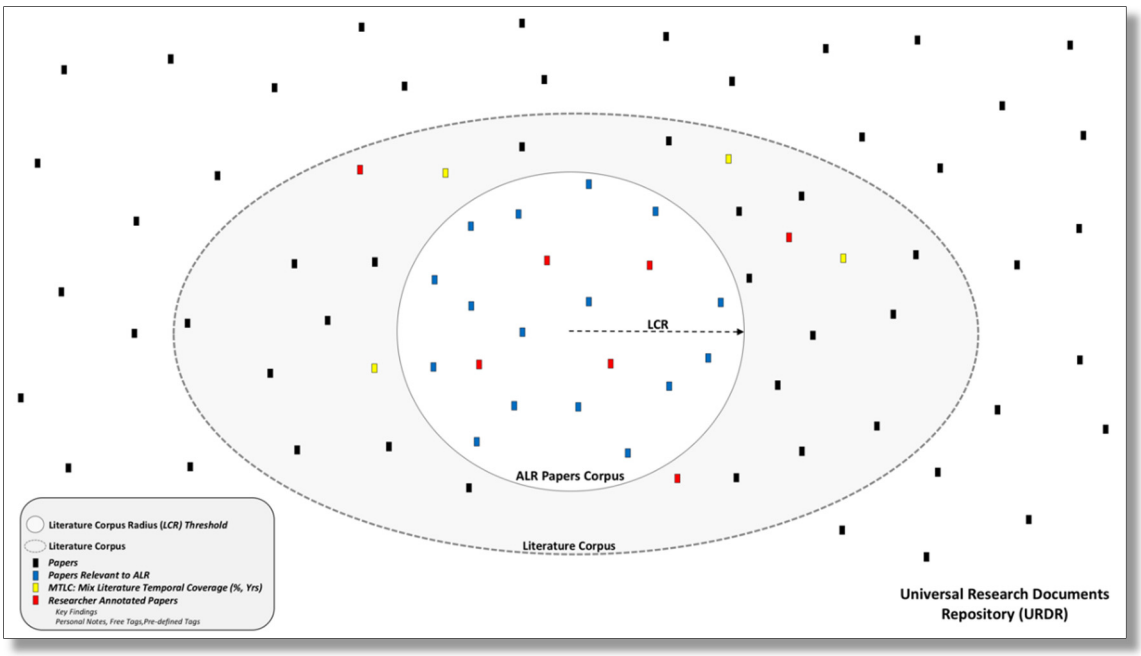


Figure 2.9 STELLAR V1 corpus representation

Specifically, STELLAR V1 computes two types of index to rank scientific papers, as shown in Appendix III:

1. LCR for literature corpus identification according to researchers' selection parameters and annotations,
2. dynamic topic based index (DTb index) for relevant papers identification.

First, a corpus of papers matching the researcher selection parameters is selected from the literature corpus. Next, based on specific researcher selection parameters, the LCR index of each paper in the previous corpus is computed and used to build a new corpus of papers. This new corpus is the set of papers whose LCR index is below a threshold defined by the researchers. STELLAR also proposes a DTb index to sort a corpus of papers or evaluate lists of references in existing literature reviews in terms of relevance for a specific research topic. For the DTb index, STELLAR considers more criteria than any other approach, such as venue age, citation category and polarity, author's impact, etc. The STELLAR V1 prototype includes the following contributions:

1. The prototype uses semantic annotations to improve document comprehension time;
2. Word co-occurrence relations across the document are used to extend topic modeling with semantic information;
3. The latent co-occurrence relations between two terms are measured from an isolated term-term perspective;
4. The prototype uses MLM and semantic relations to detect new topics automatically in multiple documents;
5. The STELLAR V1 prototype identifies and ranks relevant papers, uses citation count, and considers the age of papers, the social-level metric, as well as citation category and polarity to measure scientific research impact. It focuses on text-based analysis using metadata other than title and abstract to identify relevant papers using the researcher selection for research domain, research specific topic, matching keywords and description of research subject;
6. Scientific research papers have a specific structural organization that differentiates them from other types of documents, such as narrative texts or biographies. STELLAR

V1 adapts conventional text summarization to take into account the specificities of scientific papers in terms of document organization and rhetorical devices;

7. Finally, STELLAR V1 proposes to aggregate ALR associated objects to form a reusable Assisted Literature Research Object (ALRO).

To assist and narrow down the search results, many innovative views of the ALR have also been designed and implemented:

1. Timeline of Document-based Literature Corpus Radius,
2. Document-based Literature Corpus Radius,
3. Timeline of Author-based Literature Corpus Radius,
4. Author-based Literature Corpus Radius.

The performance of the STELLAR V1 prototype, which identifies and ranks relevant papers according to specific metadata such as topic, language, description and discipline, has been evaluated and compared to the set of documents from a baseline manual LR through a number of simulations. For this performance measurement, the volume of data was limited but is actually expanding because of the continuous harvesting of metadata from a growing number of sources in the SMESE research platform. In terms of accuracy, STELLAR V1 provides an average accuracy of 0.91 per scenario and an average precision of 0.96 per scenario; details of the simulations are shown in Appendix III.

The main primary results of this thesis are the following:

1. a rules-based harvesting and metadata-based decision support ecosystem,
2. all related algorithms to enrich metadata with topics and emotions,
3. two conceptual models and their three associated prototypes (SMESE V1 and V3 and STELLAR V1),
4. a tool to assist researchers in the building of an ALR for a specific topic or area of research.

Also, the results of this thesis included 7 published papers (as june 2nd 2017) and are described in the future works section.

FUTURE WORKS

The thesis opens up several new avenues for future research, including:

1. Summarization of Abstract of Abstracts (AoA) – AoA for scientific papers will be an extension of the current STELLAR V1. Based on a proposed scientific paper summarization technique, abstracts will be used as inputs for our summarization technique to generate the AoA of the ALR;
2. Digital Resources Metadata Enrichment (DRME) based on MLM and search engine – DRME will be a tool to aggregate metadata from content with no published metadata. It will use MLMs and a centralized search interface to discover and enrich the hidden semantic metadata related to different digital repositories of content;
3. Multi-Devices Content Machine Learning-based Assisted Recommendations, or STELLAR V2– This is an evolution of the current SMESE V3 and STELLAR V1. STELLAR V2 will use SMESE V3 as a prerequisite ecosystem. Its goal will be to match different types of content with the user's interest, emotion, availability and historical behavior.

Of the nine papers written from this thesis, seven (7) have been already published, and two (2) papers are still in evaluation and being considered for publication.

Here are the seven (7) published papers from this thesis:

1. A Semantic Metadata Enrichment Software Ecosystem (SMESE) Based on a Multi-Platform Metadata Model for Digital Libraries,


Scientific Research Publishing

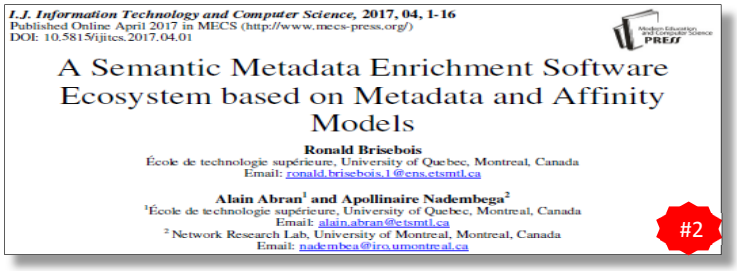
Journal of Software Engineering and Applications, 2017, 10, 370-405
http://www.scirp.org/journal/jsea
ISSN Online: 1945-3124
ISSN Print: 1945-3116

A Semantic Metadata Enrichment Software Ecosystem (SMESE) Based on a Multi-Platform Metadata Model for Digital Libraries

Ronald Brisebois[1], Alain Abran[1], Apollinaire Nadembega[2*]

[1]École de Technologie Supérieure, Université du Québec, Montréal, Canada
[2]University of Montreal, Montréal, Canada
Email: ronald.brisebois.1@ens.etsmtl.ca, alain.abran@etsmtl.ca, *apollinaire.nadembega@umontreal.ca

2. A Semantic Metadata Enrichment Software Ecosystem based on Metadata and Affinity Models


I.J. Information Technology and Computer Science, 2017, 04, 1-16
Published Online April 2017 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2017.04.01

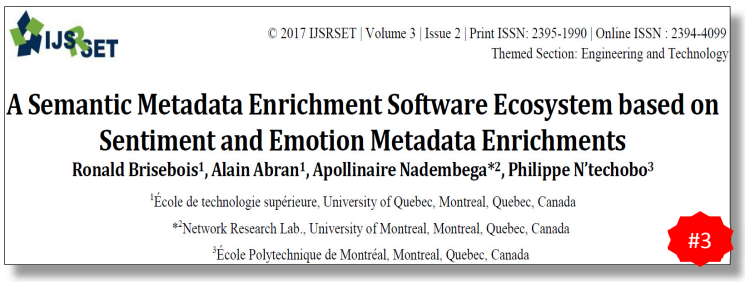
A Semantic Metadata Enrichment Software Ecosystem based on Metadata and Affinity Models

Ronald Brisebois
École de technologie supérieure, University of Quebec, Montreal, Canada
Email: ronald.brisebois.1@ens.etsmtl.ca

Alain Abran[1] and Apollinaire Nadembega[2]
[1]École de technologie supérieure, University of Quebec, Montreal, Canada
Email: alain.abran@etsmtl.ca
[2] Network Research Lab, University of Montreal, Montreal, Canada
Email: nadembea@iro.umontreal.ca

#2

3. A Semantic Metadata Enrichment Software Ecosystem base on Sentiment and Emotion Metadata Enrichments


© 2017 IJSRSET | Volume 3 | Issue 2 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

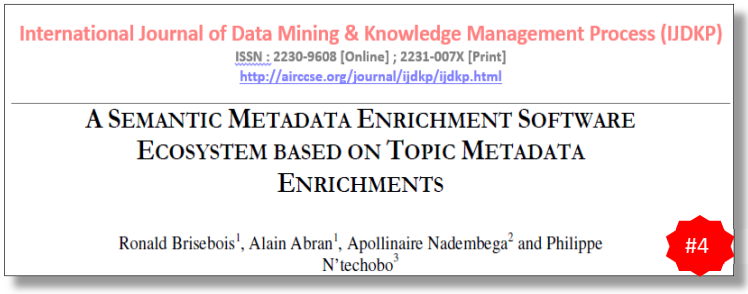
A Semantic Metadata Enrichment Software Ecosystem based on Sentiment and Emotion Metadata Enrichments

Ronald Brisebois[1], Alain Abran[1], Apollinaire Nadembega*[2], Philippe N'techobo[3]

[1]École de technologie supérieure, University of Quebec, Montreal, Quebec, Canada
*[2]Network Research Lab., University of Montreal, Montreal, Quebec, Canada
[3]École Polytechnique de Montréal, Montreal, Quebec, Canada

#3

4. A Semantic Metadata Enrichment Software Ecosystem based on Topic Metadata Enrichments

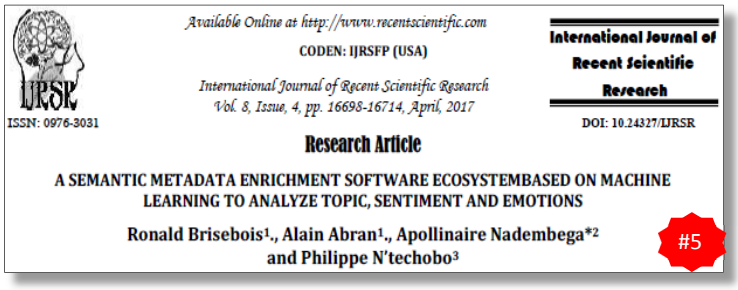

International Journal of Data Mining & Knowledge Management Process (IJDKP)
ISSN : 2230-9608 [Online] ; 2231-007X [Print]
http://airecse.org/journal/ijdkp/ijdkp.html

A SEMANTIC METADATA ENRICHMENT SOFTWARE ECOSYSTEM BASED ON TOPIC METADATA ENRICHMENTS

Ronald Brisebois[1], Alain Abran[1], Apollinaire Nadembega[2] and Philippe N'techobo[3]

#4

5. A Semantic Metadata Enrichment Software Ecosystem Based on Machine Learning to Analyse Topic, Sentiment and Emotions


Available Online at http://www.recentscientific.com
CODEN: IJRSFP (USA)
International Journal of Recent Scientific Research
Vol. 8, Issue, 4, pp. 16698-16714, April, 2017
ISSN: 0976-3031
International Journal of Recent Scientific Research
DOI: 10.24327/IJRSR

Research Article

A SEMANTIC METADATA ENRICHMENT SOFTWARE ECOSYSTEMBASED ON MACHINE LEARNING TO ANALYZE TOPIC, SENTIMENT AND EMOTIONS

Ronald Brisebois[1]., Alain Abran[1]., Apollinaire Nadembega*[2] and Philippe N'techobo[3]

#5

6. Efficient Scientific Research Literature Ranking Model based on Text and Data Mining Technique

International Journal of Engineering Research And Management (IJERM)
ISSN: 2349- 2058, Volume-04, Issue-02, February 2017

Efficient Scientific Research Literature Ranking Model based on Text and Data Mining Technique

Ronald Brisebois, Alain Abran, Apollinaire Nadembega, Philippe N'techobo

#6

7. Text and Data Mining & Machine Learning Models to Build and Assisted Literature Review with relevant papers

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

TEXT AND DATA MINING & MACHINE LEARNING MODELS TO BUILD AN ASSISTED LITERATURE REVIEW WITH RELEVANT PAPERS

Ronald Brisebois[1], Alain Abran[1], Apollinaire Nadembega[2*], Philippe N'techobo[3]
[1] École technologie supérieure, Université du Québec, Canada
[2] Network Research Lab., University of Montreal, Canada
[3] École Polytechnique de Montréal, Canada
*Corresponding author E-mail: apollinaire.nadembega@umontreal.ca

#7

Due to the large size of the three (3) technical reports proposed in this thesis, the journal editors recommended to shorten them; for this reason, nine (9) papers were prepared based on the three technical reports. Table 2.2 shows the distribution of the three technical reports into the nine papers. The full texts of each of the seven published papers are presented in annex.

Table 2.2 Distribution of the three technical report into the nine (9) papers.

Technical reports	Papers	Titles of papers	Status
1	Paper #1	A Semantic Metadata Enrichment Software Ecosystem (SMESE) based on a Multi-platform Metadata Model for Digital Libraries	Published
	Paper #2	A Semantic Metadata Enrichment Software Ecosystem based on Metadata and Affinity Models	Published
2	Paper #3	A Semantic Metadata Enrichment Software Ecosystem based on Sentiment and Emotion Metadata Enrichments	Published
	Paper #4	A Semantic Metadata Enrichment Software Ecosystem based on Topic Metadata Enrichments	Published
	Paper #5	A Semantic Metadata Enrichment Software Ecosystem based on Machine Learning to Analyse Topic, Sentiment and Emotions	Published
3	Paper #6	Efficient Scientific Research Literature Ranking Model based on Text and Data Mining Technique	Published
	Paper #7	Text and Data Mining & Machine Learning Models to Build an Assisted Literature Review with Relevant Papers	Published
	Paper #8	An Assisted Literature Review using Machine Learning Models to Recommend a Relevant Reference Papers List	* Under Review
	Paper #9	An Assisted Literature Review using Machine Learning Models to Identify and Build a Literature Corpus	* Under Review

* Verified on June 19, 2017

In the Table 2.3, we can see the journals where the papers have been published and their respective impact factor.

Table 2.3 Published papers and journal impact factors.

Number	Paper Title	Journal	Impact Factor
Paper #1	**A Semantic Metadata Enrichment Software Ecosystem (SMESE) based on a Multi-platform Metadata Model for Digital Libraries**	Journal of Software Engineering and Applications (JSEA)	2-GJIF: **1.25** RGJI: **0.5** **14th in the top 20** publications matching Software Engineering based on Google Scholar Metrics (June 2016)
Paper #2	**A Semantic Metadata Enrichment Software Ecosystem based on Metadata and Affinity Models**	International Journal of Information Technology and Computer Science (IJITCS)	GIF 2015: **0.716** ICV 2014: **8.31**
Paper #3	**A Semantic Metadata Enrichment Software Ecosystem based on Sentiment and Emotion Metadata Enrichments**	International Journal of Scientific Research in Science Engineering and Technology (IJSRSET)	SJIF 2015: **3.632** GIF 2015: **0.453**
Paper #4	**A Semantic Metadata Enrichment Software Ecosystem based on Topic Metadata Enrichments**	International Journal of Data Mining & Knowledge Management Process (IJDKP)	
Paper #5	**A Semantic Metadata Enrichment Software Ecosystem based on Machine Learning to Analyse Topic, Sentiment and Emotions**	INTERNATIONAL JOURNAL OF RECENT SCIENTIFIC RESEARCH (IJRSR)	SJIF 2016: **6.86** ICV: **5.72**
Paper #6	**Efficient Scientific Research Literature Ranking Model based on Text and Data Mining Technique**	International Journal of Engineering Research And Management (IJERM)	IF 2014-2015: **2.37**
Paper #7	**Text and Data Mining & Machine Learning Models to Build an Assisted Literature Review with Relevant Papers**	International Journal of Scientific Research in Information Systems and Engineering (IJSRISE)	GIF 2015: **0.565**

The Figure 2.10 illustrates the STELLAR V2 future works using MLMs, K Graph and NPL, with its main components.

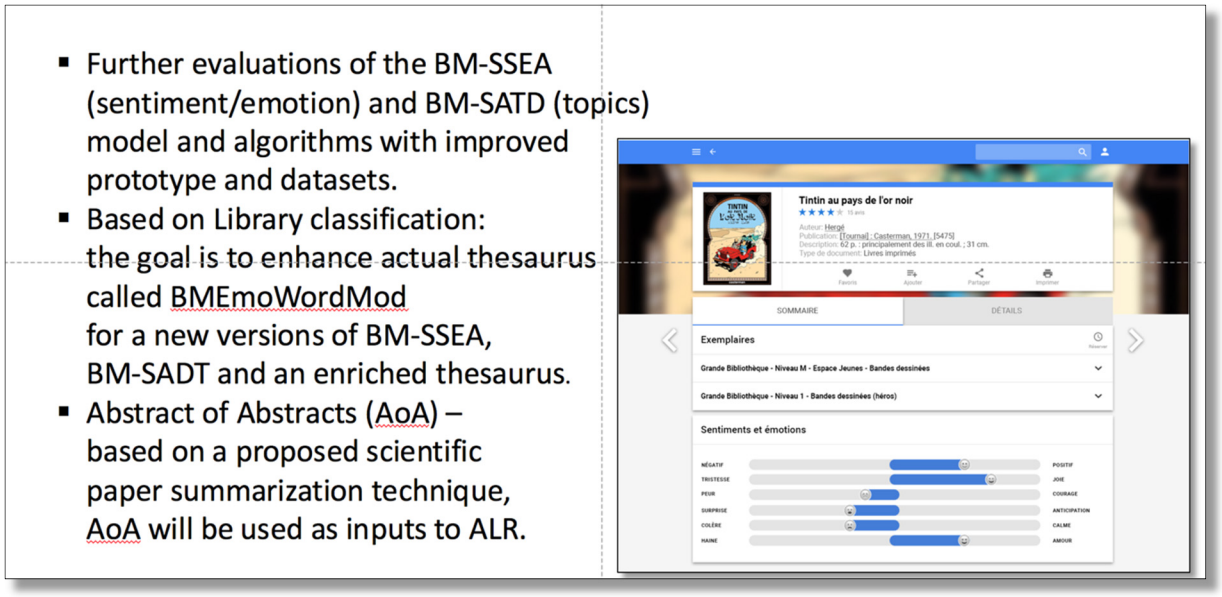


Figure 2.10 STELLAR V2 future works

STELLAR V2 will enhance the SMESE V3 prototype by adding the ability to harvest semantic metadata from different sources such as TV guides, radio program schedules, books and event calendars, and to create triple stores to define relationships enriching the metadata content. A number of additional MLMs, algorithms and prototypes will have to be developed and refined (see Figure 2.11), including:

1. An algorithm to identify the Recommended User Interest-based New Content of Events (RUINCE criteria) representing the user's evolving interests and availability;
2. An algorithm to develop analytical recommendations of subscriptions to content and events that will meet RUINCE criteria including the historical user behavior;
3. An algorithm to recommend to content or events matching user interest and emotion according to the RUINCE affinity model;
4. An algorithm to dynamically rank content or events according to the RUINCE criteria to create channels based on interests;
5. An algorithm to identify and learn interests and emotions from a multitude of human interfaces such as touchscreens, gesture interfaces, voice recognition or VR interfaces supporting navigation in STELLAR V2.

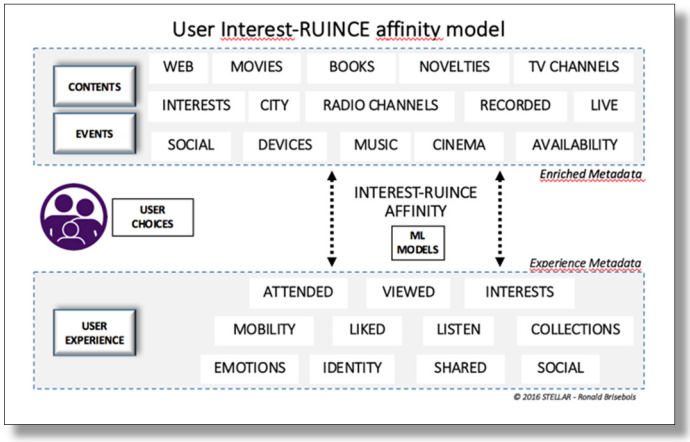


Figure 2.11 User interest-RUINCE affinity model

Furthermore, for a future version of STELLAR, we plan to work on MLM using learning process to enrich thesaurus as shown in Figure 2.12.

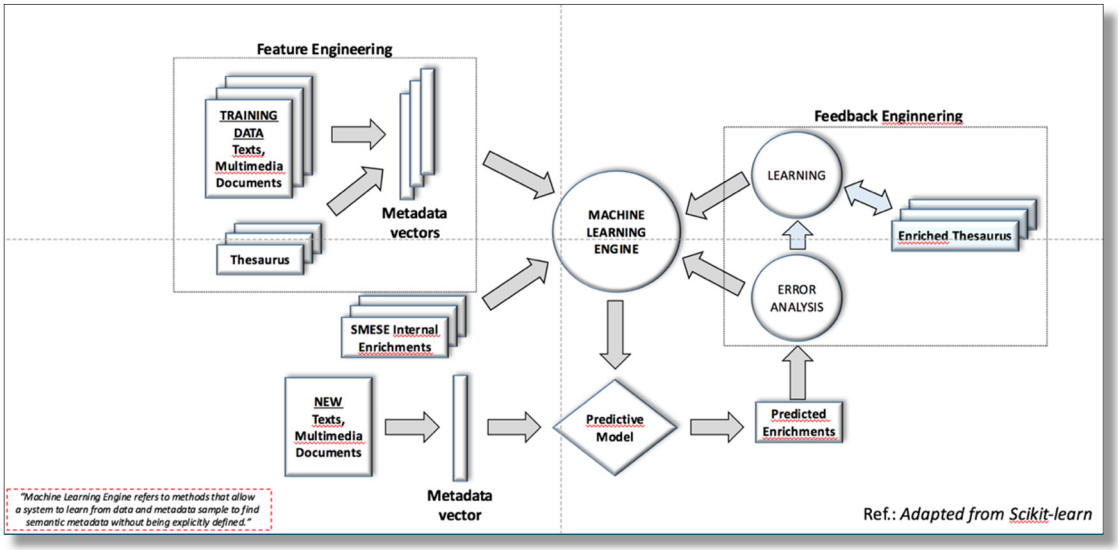


Figure 2.12 STELLAR V2 MLM – Enriched Thesaurus

APPENDIX I

A Semantic Metadata Enrichment Software Ecosystem (SMESE) Based on a Multi-platforms Metadata Model for Digital Libraries

Ronald Brisebois[1], Alain Abran[2], Apollinaire Nadembega[1]

[1] Bibliomondo, Montréal, Canada
{ronald.brisebois, apollinaire.nadembega}@bibliomondo.com
[2] École de technologie supérieure, University of Quebec, Canada,
alain.abran@etsmtl.ca

Paper submitted for publication to the International Journal for Digital Libraries,
October 2016

Abstract

Software industry has evolved to multi-product and multi-platform development based on a mix of proprietary and open source components. Such integration has occurred in software ecosystems (SECO) through a software product line engineering (SPLE) process. However, metadata are underused in the SPLE and interoperability challenge.

The proposed method is first, a semantic metadata enrichment software ecosystem (SMESE) to support multi-platform metadata driven applications, and second, based on mapping ontologies SMESE aggregates and enriches metadata to create a semantic master metadata catalogue (SMMC).

The proposed SPLE process uses a component-based software development (CBSD) approach for integrating distributed content management enterprise applications, such as digital libraries. To perform interoperability between existing metadata models (such as Dublin Core, UNIMARC, MARC21, RDF/RDA and BIBFRAME), SMESE implements an ontology mapping model. SMESE consists of nine sub-systems:

1. Metadata initiatives & concordance rules,
2. Harvesting of web metadata & data,
3. Harvesting of authority's metadata & data,
4. Rule-based semantic metadata external enrichment,
5. Rule-based semantic metadata internal enrichment,
6. Semantic metadata external & internal enrichment synchronization,
7. User interest-based gateway,
8. Semantic master catalogue,
9. Semantic analytical.

To conclude, this paper proposes a decision support process, called SPLE decision support process (SPLE-DSP) which is then used by SMESE to support dynamic reconfiguration. SPLE-DSP consists of a dynamic and optimized metadata-based reconfiguration model (DOMRM). SPLE-DSP takes into account runtime metadata-based variability functionalities, context-awareness and self-adaptation. It also presents the design and implementation of a working prototype of SMESE applied to a semantic digital library.

Keywords: Digital library, metadata enrichment, semantic metadata enrichment, software ecosystem, software product line engineering.

1. Introduction

With more and more data available on the web, how users search and discover contents is of crucial importance. There is growing research on interaction paradigms investigating how users may benefit from the expressive power of semantic web standards.

The semantic web may be defined as the transformation of the world wide web to a database of linked resources, where data may be widely reused and shared (Lacasta et al., 2013). Web services can be enhanced by drawing on semantically aware data made available by a variety of providers. In addition, as information discovery needs become more and more challenging

traditional keyword-based information retrieval methods are increasingly falling short in providing adequate support. This retrieval problem is compounded by the poor quality of the metadata content in some digital collections.

SECO (Albert, Santos, & Werner, 2013; Amorim, Almeida, & McGregor, 2013; Christensen et al., 2014; Di Ruscio et al., 2014; dos Santos, Esteves, Freitas, & de Souza, 2014; Ghapanchi, Wohlin, & Aurum, 2014; Henderson-Sellers, Gonzalez-Perez, McBride, & Low, 2014; Jansen & Bloemendal, 2013; Lim, Bentley, Kanakam, Ishikawa, & Honiden, 2015; Manikas & Hansen, 2013; Mens, Claes, Grosjean, & Serebrenik, 2014; Musil, Musil, & Biffl, 2013; Park & Lee, 2014; Robillard & Walker, 2014; Shinozaki et al., 2015; Urli, Blay-Fornarino, Collet, Mosser, & Riveill, 2014) is defined as the interaction of a set of actors on top of a common technological platform providing a number of software solutions or services (Christensen et al., 2014; Manikas & Hansen, 2013). In SECO, internal and external actors create and compose relevant solutions together with a community of domain experts and users to satisfy customer needs within specific market segments. This poses new challenges since the software systems providing the technical basis of a SECO are being evolved by various distributed development teams, communities and technologies.

There is growing agreement for the general characteristics of SECO, including a common technological platform enabling outside contributions, variability-enabled architectures, tool support for product derivation, as well as development processes and business models involving internal and external actors. At least ten SECO characteristics have been identified (Lettner et al., 2014) that focus on technical processes for development and evolution - see Table A 1.1.

Table A 1.1 SECO characteristics
Taken from (Lettner et al., 2014)

1	Internal and external developers
2	Evaluative common technological platform
3	Controlled central part
4	Enable outside contributions and extensions
5	Variability-enabled architecture
6	Shared core assets
7	Automated and tool-supported product derivation
8	Outside contributions included in the main platform
9	Tools, frameworks and patterns
10	Distribution channel

Gawer and Cusumano (Gawer & Cusumano, 2014) have analyzed a wide range of industry examples of SECO and identified two predominant types of platforms:

1. Internal platforms (company or product): defined as a set of assets organized in a common structure from which a company can efficiently develop and produce a stream of derivative products;
2. External platforms (industry): defined as products, services, or technologies that act as a foundation upon which external innovators, organized as an innovative business ecosystem, can develop their own complementary products, technologies, or services.

Indeed, the new generation of SECO must be an integration of multi-platforms (internal and external) that allows the interaction of a set of internal and external actors.

Concurrently modern software demands more and more adaptive features, many of which must be performed dynamically. In this context, a collaborative platform is important in order to coordinate collaborative and distributed environments for development of SECO platforms.

Furthermore, as the requirement of SECO to support adaptation capabilities of systems is increasing in importance (Andrés et al., 2013) it is recommended such adaptive features be included within software product lines (SPL) (Capilla et al., 2014; Harman et al., 2014; Metzger & Pohl, 2014; Olyai & Rezaei, 2015). The SPL concept is appealing to organizations dealing with software development that aims to provide a comprehensive model for an

organization building applications based on a common architecture and core assets (Andrés et al., 2013; Metzger & Pohl, 2014).

SPLs have been used successfully in industry for building families of systems of related products, maximizing reuse, and exploiting their variable and configurable options (Harman et al., 2014).

SPL development can be divided into three interrelated activities:

1. Core assets development: may include architecture, reusable software components, domain models, requirement statements, documentation, schedules, budgets, test plans, test cases, process descriptions, modeling diagrams, and other relevant items used for product development;
2. Product development: represents activities where products are physically developed from core assets, based on the production plan, in order to satisfy the requirements of the SPL (Krishnan, Strasburg, Lutz, Goseva-Popstojanova, & Dorman, 2013);
3. Management: involves the essential processes carried out at technical and organizational levels to support the SPL process and ensures that the necessary resources are available and well-coordinated.

To develop and implement SPL the literature proposes several SPL frameworks (Olyai & Rezaei, 2015) using a variety of CBSD approaches (Quadri & Abubakar, 2015; Singh, Sangwan, Singh, & Pratap, 2015; Yadav & Yadav, 2015):

1. COPA (component-oriented platform architecting): an SPL framework that is component-oriented;
2. FAST (family-oriented abstraction, specification and translation): a software development process that divides the process of a product line into three sections: domain qualification, domain engineering and application engineering;
3. FORM (feature-oriented reuse method): a feature-oriented method that, by analyzing the features of the domain, uses these features to provide the SPL architecture. FORM focuses on capturing commonalities and differences of applications in a domain in terms of features and uses the analysis results to develop domain architectures and components;

4. Kobra: a component-oriented approach based on the UML features that integrate the two paradigms into a semantic, unified approach to software development and evolution;
5. QADA (quality-driven architecture design and analysis): a product line architecture design method that provides traceability between the product quality and design time quality assessment.

Semantic web (Jeremić et al., 2013; Khriyenko & Nagy, 2011; Lécué et al., 2014; Ngan & Kanagasabai, 2013; Rettinger et al., 2012) linked data is the most important concept to support Semantic Metadata Enrichment (SME) in a SECO architecture (Aleti, Buhnova, Grunske, Koziolek, & Meedeniya, 2013; Capilla, Jansen, Tang, Avgeriou, & Babar, 2016; Demir, 2015; Ginters, Schumann, Vishnyakov, & Orlov, 2015; Neves, Carvalho, & Ralha, 2014; Oussalah, Bhat, Challis, & Schnier, 2013; Yang, Liang, & Avgeriou, 2016).

Today, semantic web technologies, for example in digital libraries, offer a new level of flexibility, interoperability and a way to enhance peer communication and knowledge sharing by expanding the usefulness of the digital libraries that in the future will contain the majority of data. Indeed, a semantic web TDM, based on semantic web technology, ensures more closely relevant results based on the ability to understand the definition and user-specific meaning of the word or term being searched for. Semantic search of semantic web engines are better able to understand the context in which the words are being used, resulting in relevant results with greater user satisfaction. Unfortunately, in the public domain there is a scarcity of search engines that follow a semantic-based approach to searching and browsing data (Ngan & Kanagasabai, 2013). Furthermore, the web is currently not contextually organized.

Thus, to enrich web data by transforming it into knowledge accessible by users, we propose a multi-platform architecture, referred to as SMESE, which uses a CBSD approach to integrate distributed content management enterprise applications, such as libraries and the Software Product Line Engineering (SPLE) approach.

Our SMESE architecture includes mobile first design (MFD) and semantic metadata enrichment (SME) engines that consist of metadata and meta-entity enrichment based on mapping ontologies and a semantic master metadata catalogue (SMMC).

More specifically, our SMESE implements a new decision support process in the context of SPLE, called the SPLE decision support process (SPLE-DSP), a meta entity model that represents all library materials and a meta metadata model. SPLE-DSP allows support for metadata-based reconfiguration. It consists of a dynamic and optimized metadata based reconfiguration model (DOMRM) where users select their preferences in the market place.

The major contributions of this paper are:

1. Definition of a software ecosystem model that configures the application production process including software aspects based on a proposed CBSD and metadata-based SPLE approach;
2. Definition and partial implementation of semantic metadata enrichment using SPLE and a semantic master metadata catalogue (SMMC) to create a universal metadata knowledge gateway (UMKG);
3. Design and implementation of a SMESE prototype for a semantic digital library (Libër).

This paper proposes a semantic metadata enrichment software ecosystem (SMESE) to support multi-platform metadata driven applications, such as a semantic digital library. Based on mapping ontologies SMESE also integrates and enriches data and metadata to create a semantic master metadata catalogue (SMMC).

The remainder of the paper is organized as follows. Section 2 is a literature review. Section 3 presents the multi-platform architecture of the proposed SMESE, and Section 4, the related nine sub-systems. Section 5 presents the prototype of a SMESE implementation in an industry context. Section 6 presents a summary and ideas for future work.

2. Literature review

A software product line (SPL) (Andrés et al., 2013; Ayala, Amor, Fuentes, & Troya, 2015; Capilla et al., 2014; Harman et al., 2014; Horcas, Pinto, & Fuentes, 2016; Krishnan et al., 2013; Metzger & Pohl, 2014; Olyai & Rezaei, 2015) is a set of software intensive systems that share a common and managed set of features satisfying the specific needs of a particular market segment developed from a common set of core assets in a prescribed way (Metzger & Pohl, 2014; Olyai & Rezaei, 2015). SPL engineering aims at: effective utilization of software assets, reducing the time required to deliver a product, improving quality, and decreasing the cost of software products.

The following sub-sections present the four research axes related to our research:

1. Software product line engineering (SPLE),
2. SECO architecture using component integration and component evolution,
3. SECO architecture and SPLE,
4. Semantic metadata enrichment (SME).

The related works section is at the intersection of SPLE, service-oriented computing, cloud computing, semantic metadata and adaptive systems.

2.1 Software product line engineering (SPLE)

The development of software involves requirements analysis, design, construction, testing, configuration management, quality assurance and more, where stakeholders always look for high productivity, low cost and low maintenance. This has led to software product line engineering (SPLE) (Capilla et al., 2014) as a comprehensive model that helps software providers to build applications for organizations/clients based on a common architecture and core assets. SPLE deals with the assembly of products from current core assets, commonly known as components, within a component-based architecture (W. He & Xu, 2014; Mück & Fröhlich, 2014), and involves the continuous growth of the core assets as production proceeds.

Note that the following related works are organized according to two axes: organizational and technical.

An overview of SPLE challenges is presented in (Capilla et al., 2014; Harman et al., 2014; Metzger & Pohl, 2014). Metzger and Pohl (Metzger & Pohl, 2014) suggest that the successful introduction of SPLE heavily depends on the implementation of adequate organizational structures and processes. They also identify three trends expected from SPLE research in the next decade:

1. managing variability in non-product-line settings,
2. leveraging instantaneous feedback from big data and cloud computing during SPLE,
3. addressing the open world assumption in software product line settings.

A survey of works on search based software engineering (SBSE) for SPLE is presented in Harman et al. (Capilla et al., 2014; Harman et al., 2014).

Capilla et al. (Capilla et al., 2014) provide an overview of the state of the art of dynamic software product line architectures and identify current techniques that attempt to tackle some of the many challenges of runtime variability mechanisms. They also provide an integrated view of the challenges and solutions that are necessary to support runtime variability mechanisms in SPLE models and software architectures. According to them, the limitations of today's SPLE models are related to their inability to change the structural variability at runtime, provide the dynamic selection of variants, or handle the activation and deactivation of system features dynamically and/or autonomously. SPLE is, therefore, the natural candidate within which to address these problems. Since it is impossible to predict all the expected variability in a product line, SPLE must be able to produce adaptable software where runtime variations can be managed in a controlled manner. Also, to ensure performance in systems that have strong real-time requirements, SPLE must be able to handle the necessary adaptations and current reconfiguration tasks after the original deployment due to the computational complexity during variants selection.

Olyai and Rezaei (Olyai & Rezaei, 2015) describe the issues and challenges surrounding SPLs, introduce some SPLE ecosystems and compare them, based on the issues and challenges, with

a view to how each ecosystem might be improved. The issues and challenges are presented in terms of administrative and organizational aspects and technical aspects. The administrative and organizational comparison criteria include strategic plans of the organization while the technical comparison criteria include requirements, design, implementation, test and maintenance. According to them, there is not a single approach that takes into account all these criteria together. Also, no single approach takes into account metadata for implementation and testing.

2.2 SECO architecture using components integration and components evolution

Software ecosystems (SECO) (Aleti et al., 2013; Capilla et al., 2016; Christensen et al., 2014; Gawer & Cusumano, 2014; Manikas & Hansen, 2013; Mens et al., 2014; Shinozaki et al., 2015) consist of multiple software projects, often interrelated to each other by means of dependency relationships. When one project undergoes changes and issues a new release, this may or may not lead other projects to upgrade their dependencies. Unfortunately, the upgrade of a component may create a series of issues. In their systematic literature review of SECO research, Manikas and Hansen (Manikas & Hansen, 2013) report that while research on SECO is increasing:

1. There is little consensus on what constitutes a SECO;
2. Few analytical models of SECO exist;
3. Little research is done in the context of real-world SECO.

They define a SECO as the interaction of a set of actors on top of a common technological platform that results in a number of software solutions or services where each actor is motivated by a set of interests or business models while connected to the rest of the actors. They also identify three main components of SECO architecture:

1. SECO software engineering: focuses on technical issues related directly or indirectly to the technological platform of a SECO;
2. SECO business and management: focuses on the business, organizational and management aspects of a SECO;

3. SECO relationships: represent the social aspect of SECO architecture since it is essential for SPLE actors to interact among themselves and with the platform.

2.3 SECO architecture and SPLE

This section focuses on SECO architecture related to SPLE, beginning with an industry perspective.

Christensen et al. (Christensen et al., 2014) define the concept of SECO architecture as a set of structures comprised of actors and software elements, the relationships among them, and their properties. They present the Danish telemedicine SECO in terms of this concept, and discuss challenges that are relevant in areas beyond telemedicine. They also discuss how software engineering practice is affected by describing the creation and evolution of a central SECO architecture, namely Net4Care, that serves as a reference architecture and learning vehicle for telemedicine and for the actors within a single software organization.

Demir (Demir, 2015) also proposes a software architecture that is strongly related to a defence system and limited to military personnel. Their multi-view SECO architecture design is described step by step. They begin by identifying the system context, requirements, constraints, and quality expectations, but do not describe the end products of the SECO architecture. They also introduce a novel architectural style, called “star-controller architectural style” (Demir, 2015) where synchronization and control of the flow of information are handled by controllers. However, a major drawback of this style is that failure of one controller disables all the subcomponents attached to that controller.

Neves et al. (Neves et al., 2014) propose an architectural solution based on ontology and the spreading algorithm that offers personalized and contextualized event recommendations in the university domain. They use an ontology to define the domain knowledge model and the spreading activation algorithm to learn user patterns through discovery of user interests. The main limitation of their architectural context-aware recommender system is that it is specific to university populations and does not present the actual model of the system that shows the interactions between the components and the data.

Alferez et al. (Alférez, Pelechano, Mazo, Salinesi, & Diaz, 2014) propose a framework that uses semantically rich variability models at runtime to support the dynamic adaptation of service compositions. They argue that should problematic events occur, functional pieces may be added, removed, replaced, split or merged from a service composition at runtime, hence delivering a new service composition configuration. Based on this argument, they propose that service compositions be abstracted as a set of features in a variability model. They define a feature as a logical unit of behavior specified by a set of functional and non-functional requirements. Thus, they propose adaptation policies that describe the dynamic adaptation of a service composition in terms of the activation or deactivation of features in the causally connected variability model. Unfortunately, this variability model is limited to activation and deactivation of services. Indeed, the model should allow adaptation of services or include a service interoperability protocol (SIP) rather than compositions only according to changes in the computing infrastructure.

In component based software development (CBSD), the fuzzy logic approach (Singh et al., 2015; Yadav & Yadav, 2015) is largely used to select components. Singh et al. (Singh et al., 2015) explored the various measures such as separation of concerns (SoC), coupling, cohesion, and size measure that affect the reusability of aspect oriented software. The main drawback of their contribution is that the fuzzy logic rules are static. They do not propose a way to improve the rules based on developer satisfaction of the fuzzy inference system (FIS) output. In addition, their fuzzy inference system is limited to reusability of software.

2.4 Semantic metadata enrichment (SME)

Bontcheva et al. (Bontcheva et al., 2015) investigate semantic metadata automatic enrichment and search methods. In particular, the benefits of enriching articles with knowledge from linked open data resources are investigated with a focus on the environmental science domain. They also propose a form-based semantic search interface to facilitate environmental science researchers in carrying out better semantic searches. Their proposed model is limited to linking terms with DBpedia URI and does not take into account the semantic meaning of terms in order to detect the best DBpedia URI.

Some authors focus their enrichment model on person mobility trace data (Fileto, Bogorny, et al., 2015; Fileto, May, et al., 2015; Krueger et al., 2015; Kunze & Hecht, 2015). Krueger et al. (Krueger et al., 2015) show how semantic insights can be gained by enriching trajectory data with place of interest (POI) information using social media services. They handle semantic uncertainties in time and space, which result from noisy, imprecise, and missing data, by introducing a POI decision model in combination with highly interactive visualizations. However, this model is limited to POI detection.

Kunze and Hecht (Kunze & Hecht, 2015) propose an approach to processing semantic information from user-generated OpenStreetMap (OSM) data that specifies non-residential use in residential buildings based on OSM attributes, so-called tags, which are used to define the extent of non-residential use.

Our conclusions from these related works are:

1. SPLE architecture needs to be flexible and meet administrative and organizational aspects such as the organization's strategic plans and marketing strategies, as well as technical aspects such as requirements, design, implementation, test and maintenance;
2. Researchers need to focus on real-world SECO;
3. Several proposed SECO models do not take into account autonomic mechanisms to guide the self-adaptation of service compositions according to changes in the computing infrastructure;
4. In CBSD fuzzy inference systems (FIS) have been employed to develop the components selection model, however, there is no FIS based model that proposes more than one software measure as FIS output;
5. There is no SECO architecture that takes into account several semantic enrichment aspects;
6. Current metadata and entity enrichment models are limited to only one domain for their semantic enrichment process and therefore do not involve several enriched metadata and entity models;
7. Current metadata and entity enrichment models only link terms and DBpedia URI;

8. Current metadata and entity enrichment models do not take into account person mobility trace data gathering and analysis in the enrichment process of metadata.

3. SMESE multi-platform architecture

This section presents the proposed semantic enriched metadata software ecosystem (SMESE) architecture based on SPLE and CBSD approaches to support metadata and entity social and semantic enrichment for semantic digital libraries and based on an MFD approach for user interface design. Each component of the SMESE architecture is based on existing approaches (SPLE and CBSD) and an SME concept (proposed in this work) to generate, extract, discover and enrich metadata based on mapping ontologies and making use of contents and linked data analysis.

This section first presents an overview of the proposed SMESE multi-platform architecture followed by detailed explanations.

3.1 Overview of the proposed SMESE multi-platform model

For the new generation of information and data management, metadata is a most efficient material for data aggregation. For example, it is easier to find a specific set of interests for users based on metadata such as content topics, or based on the sentiments expressed in a content. Furthermore, it is possible to increase user satisfaction by reducing the user interest gap. To make this feasible, all content needs to be enriched. In other words, specific metadata must be available including semantic topics, sentiments and abstracts. However, at the present time more than 85% of content does not have this metadata.

The SMESE multiplatform prototype implemented at BiblioMondo, a supplier of software digital libraries, includes a process to aggregate multiple world catalogues from libraries, universities, Bbookstores, #tag collections, museums, and cities. The collection of pre-harvested and processed metadata and full text comprises the searchable content.

Central indexes typically include: full text and citations from publishers, full text and metadata from open source collections, full text, abstracting, and indexing from aggregators and subscription databases, and different formats (such as MARC) from library catalogues, also called the base index, unified index, or foundation index.

The SMESE multiplatform framework must link bibliographic records and semantic metadata enrichments into a digital world library catalogue. SMESE must search and discover actual collections or novelties, including: works, books, DVDs, CDs, comics, games, pictures, videos peoples, legacy collections, organizations, rewards, TVs, radios, and museums.

Figure A 1.1 presents the five levels of the semantic collaborative gateway:

1. MetaEntity (black),
2. Entity (blue),
3. Semantic metadata enrichment and creation (red),
4. Free sources of metadata (yellow) and subscription-based metadata,
5. Content (green).

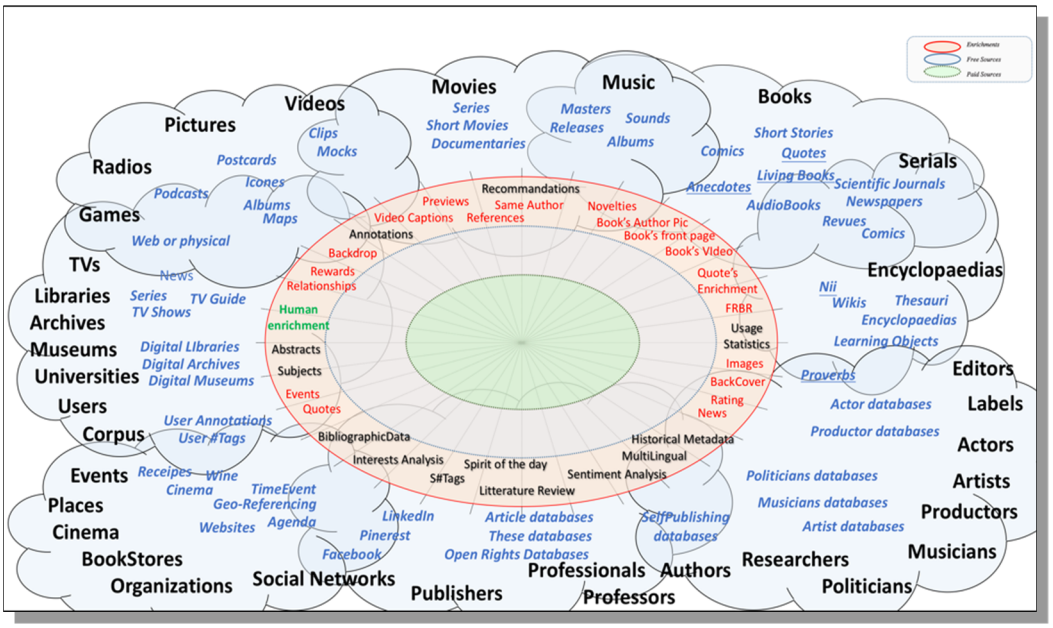


Figure A 1.1 Universal MetaModel and Metadata Enrichment

Figure A 1.2 presents the entity matrix. The metadata are defined once and are related to each specific entity.

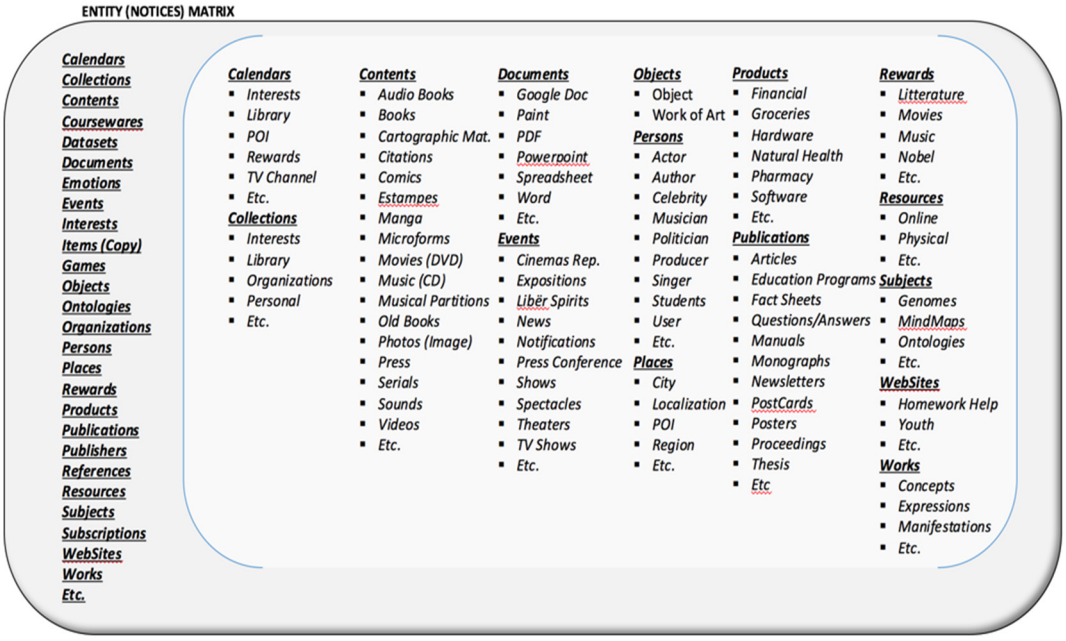


Figure A 1.2 Entity Matrix

Semantic relationships between the contents, persons, organization and places are defined and curated in the master metadata catalogue. Topics, sentiments and emotions must be extracted automatically from the contents and their context:

1. Libraries spend a lot of money buying books and electronic resources. Enrichment uncovers that information and makes it possible for people to discover the great resources available everywhere;
2. The average library has hundreds of thousands of catalogue records waiting to be transformed into linked data, turning those thousands of records into millions of relationships;
3. FRBR (functional requirements for bibliographic records) is a semantic representation of the bibliographic record. A work is a high-level description of a document, containing information such as author (person), title, descriptions, subjects, etc.,

common to all expressions, format and copy of the work. (See Figure A 1.3 for an FRBR framework description).

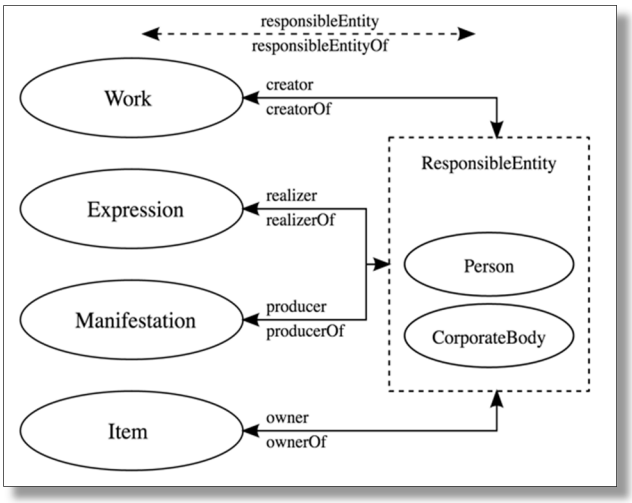


Figure A 1.3 FRBR framework description

SMESE must allow users to find topically related content through an interest-based search and discovery engine. Transforming bibliographic records into semantic data is a complex problem that includes interpreting and transforming the information. Fortunately, many international organizations (e.g., BNF, Library of Congress and some others) have partly done this heavy work and already have much bibliographic metadata converted into triple-stores.

Recent catalogues support the ability to publish and search collections of descriptive entities (described by a list of generic metadata) for data, content, and related information objects. Metadata in catalogues represent resource characteristics that can be indexed, queried and displayed by both humans and software. Catalogue metadata are required to support the discovery and notification of information within an information community. Using the information from these Semantic Metadata Enrichments, the search engine, discovery engine and notification engine are able to give to the final user better results in accord with his interest or mood.

SMESE must also include an automated approach for semantic metadata enrichment (SME) that allows users to perform interest-based semantic search or discovery more efficiently. To summarize, our SMESE makes the following contributions:

- Definition and development of a proposed semantic metadata enrichment software ecosystem. (See Figure A 1.4 SMESE overview and Figure A 1.21 SMESE detailed.

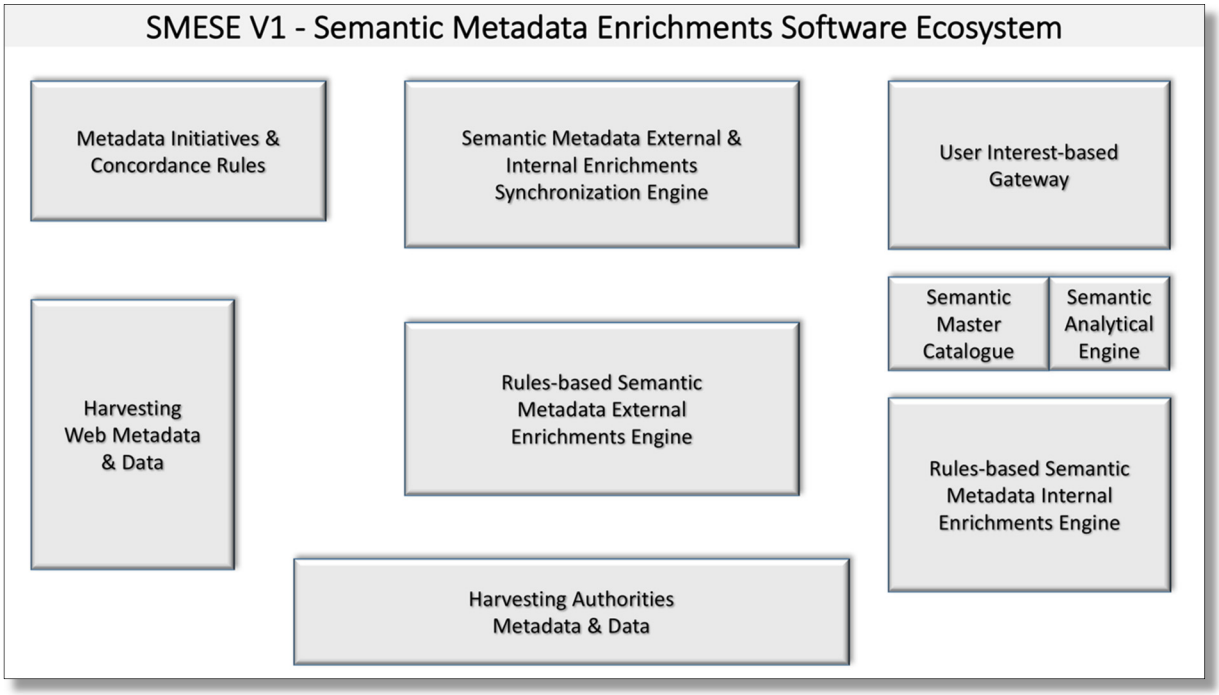


Figure A 1.4 Semantic Enriched Metadata Software Ecosystem (SMESE) Architecture

This new semantic ecosystem will harvest and enrich bibliographic records externally (from the web) and internally (from text data). The main components of the ecosystem will be:

1. Metadata initiatives & concordance rules,
2. Harvesting web metadata & data,
3. Harvesting authority's metadata & data,
4. Rule-based semantic metadata external enrichment,
5. Rule-based semantic metadata internal enrichment,
6. Semantic metadata external & internal enrichment synchronization,
7. User interest-based gateway,

8. Semantic master catalogue,
9. Semantic analytical.

- *Topic detection/generation* - A prototype was developed to automate the generation of topics from the text of a document using our algorithm BM-SATD (BiblioMondo-Semantic Annotation-based Topic Detection). In this research prototype, the following issues were investigated:

1. Semantic annotations can improve the processing time and comprehension of the document;
2. Extending topic modeling into account co-occurrence to combine semantic relations and co-occurrence relations to complement each other;
3. Since latent co-occurrence relations between two terms cannot be measured in an isolated term-term view, the context of the term must be taken into account;
4. Use of machine learning techniques to allow the ecosystem SMESE to be able to find a new topic itself.

- *Sentiment and Emotion Analysis* - The prototype developed has the following characteristics:

1. Traditional sentiment analysis methods mainly use terms and their frequency, parts of speech, rules of opinion and sentiment shifters; but semantic information is ignored in term selection;
2. Our contribution to sentiment analysis includes emotions;
3. The human contribution to improve the accuracy of our approach is taken into account.
4. Sentiment and emotion analysis are combined;
5. It is important to identify the sentiment and emotion of a book taking into account all the books of the collection;
6. The collection of documents and paragraphs are taken into account. In terms of granularity, most of the existing approaches are sentence-based;
7. These approaches did not take into account the surrounding context of the sentence which may cause some misunderstanding with discovery of sentiment and emotion. In our approach, the surrounding context of the sentence is included.

The prototype makes use of the proposed algorithm BM-SSEA (BiblioMondo-Semantic Sentiment and Emotion Analysis). The SMEE algorithm fulfills all the attributes of Table A 1.1.

The SMESE extends the SECO characteristics presented in (Lettner et al., 2014) from 10 to 12. See Table A 1.1 SECO characteristics versus Table A 1.2 SMESE characteristics.

Table A 1.2 SMESE characteristics

1	Internal and external developers
2	Evaluative common technological platform
3	Controlled central part
4	Enable outside contributions and extensions
5	Variability-enabled architecture
6	Shared core assets
7	Automated and tool-supported product derivation
8	Outside contributions included in main platform
9	Social network and IoT integration
10	Semantic Metadata Internal Enrichments
11	Semantic Metadata External Enrichments
12	User Interest-based Gateway

More specifically, the proposed SPLE approach is a combination of feature-oriented reuse method FORM and component-oriented platform architecting (COPA) approaches focusing on data and metadata enrichment. Through the combination of these two approaches, the following can be taken into account:

1. Administrative and organizational aspects such as roles and responsibilities, intergroup communication capabilities, personnel training, adoption of new technologies, strategic plans of the organization and marketing strategies;
2. Technical aspects such as requirements, design, implementation, test and maintenance.

With respect to CBSD, our SMESE includes a method for selecting composer components for design of an SPLE. This method can manage and control the complexities of the component selection problem in the creation of the declared product line. Also, the SMESE architecture supports runtime variability and multiple and dynamic binding times of products.

4. Subsystems within the SMESE multi-platform architecture

The following sub-sections present in more detail the nine subsystems designed for the prototype of this SMESE architecture.

4.1 Metadata initiatives & concordance rules (MICR)

This section presents the details of the metadata initiatives & concordance rules (MICR), specifically the semantic metadata meta-catalogue (SMMC) as shown in Figure A 1.2.

Metadata is structured information that describes, explains, locates, accesses, retrieves, uses, or manages an information resource of any kind. Metadata refers to data about data. Some use it to refer to machine understandable information, while others employ it only for records that describe electronic resources. In the library ecosystem, metadata is commonly used for any formal scheme of resource description, applying to any type of object, digital or non-digital. Many metadata schemes exist to describe various types of textual and non-textual objects including published books, electronic documents, archival documents, art objects, educational and training materials, scientific datasets and, obviously, the web.

Libraries and information centers are the intermediaries between the information, information sources and users. In order to make information accessible, libraries perform several activities, one of the most important and fundamental of which is cataloguing. The technological developments of the past 25 years have radically transformed both the process of cataloguing and access to information through catalogues.

Several rules have been proposed to cover the description and provision of access points for all library materials (entities). These rules are based on an individual framework for the

description of library materials. There is no ecosystem that allows the creation of universal, understandable and readable, metadata, that would describe all entities used in a library.

The most popular metadata models are:

1. Dublin Core (DC): primarily designed to provide a simple resource description format for networked resources. DC does not have any coding to provide the necessary details for the specification of a record that could be converted to any machine readable coding like UNIMARC, MARC21;
2. UNIMARC: consists of data formulated by highly controlled cataloguing codes. This format is difficult to understand and unreadable for the end user. For this reason, MARC21 was proposed;
3. MARC21: is both flexible and extensible and allows users to work with data in ways specific to individual library needs. MARC21 remains difficult to understand, however;
4. RDF/RDA: mainly in Europe, is a new model that includes FRBRized Bibliographic Records;
5. BIBFRAME: mainly in North America, is a new model that includes FRBRized Bibliographic Records.

In addition, there is no mapping model among these that would make them interoperable. The overall challenge is to develop: (1) a modeling of partial international standardization of entities, (2) a modeling of partial international standardization of metadata, and (3) a modeling of partial international standardization of metadata mapping ontology.

Unfortunately, the power of metadata is limited: indeed, large national and international projects of digital libraries, such as Europeana and the Digital Public Library of America, have highlighted the importance of sharing metadata across silos. While both of these projects have been successful in harvesting collections data, they have had problems with rationalizing the data and forming a coherent and semantic understanding of the aggregation.

In addition, organizations create digital collections and generate metadata in repository silos. Generally such metadata does not:

1. Connect the digitized items to their analogue sources,

2. Connect names to authority records (persons, organizations, places, etc.) nor subject descriptions to controlled vocabularies,
3. Connect to related online items accessible elsewhere.

Aggregators harvest this metadata that, in the process, generally becomes inaccurate. In fact, aggregators usually ignore idiosyncratic use of metadata schemas and enforce the use of designated metadata fields.

Connecting data across silos would help improve the ability of users to browse and navigate related entities without having to do multiple searches in multiple portals. The proposed model defines crosswalks that create pathways to different sources; each pathway checks the structure of the metadata source and then performs data harvesting. Figure A 1.5 shows the SMMC model that addresses this issue.

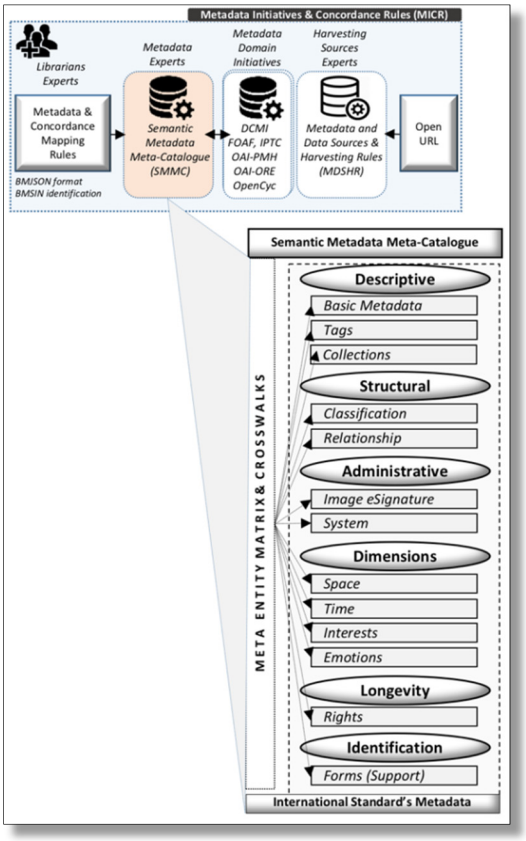


Figure A 1.5 Semantic metadata meta-catalogue (SMMC)

In SMESE the metadata is classified into six categories:

1. *Descriptive metadata*: describes and identifies information resources at the local (system) level to enable searching and retrieving (e.g., searching an image collection to find paintings of animals) at the web-level, and to enable users to discover resources (e.g., searching the web to find digitized collections of poetry). Such metadata includes unique identifiers, physical attributes (media, dimensions, conditions) and bibliographic attributes (title, author/creator, language, keywords);
2. *Structural metadata*: facilitates navigation and presentation of electronic resources and provides information about the internal structure of resources (including page, section, chapter numbering, indexes, and table of contents) in order to describe relationships among materials (e.g., photograph B was included in manuscript A), and to bind the related files and scripts (e.g., File A is the JPEG format of the archival image File B);
3. *Administrative metadata*: facilitates both short-term and long-term management and processing of digital collections and includes technical data on creation and quality control, rights management, access control and usage requirements;
4. *Dimension, longevity and identification metadata*: are new classifications that aim to increase user satisfaction, in terms of expected interests and emotions. For example, dimension metadata regroups all metadata about space, time, emotions and interests. This metadata allows finding specific content. Another example: emotions may suggest specific content to a particular user at a specific time and place. Furthermore, the source metadata identifies the provenance and the rights relative to the creation of the metadata.

4.2 Harvesting of web metadata & data (HWMD)

The harvesting of web metadata & data (HWMD) sources such as (see Figure A 1.6):

1. Semantic digital resources,
2. Digital resources,
3. Portal/websites events,
4. Social networks & events,

5. Enrichment repositories,
6. Discovery repositories,
7. Collaborative MediaLab.

The integration of these sources in SMESE allows users to aggregate and enrich metadata and data.

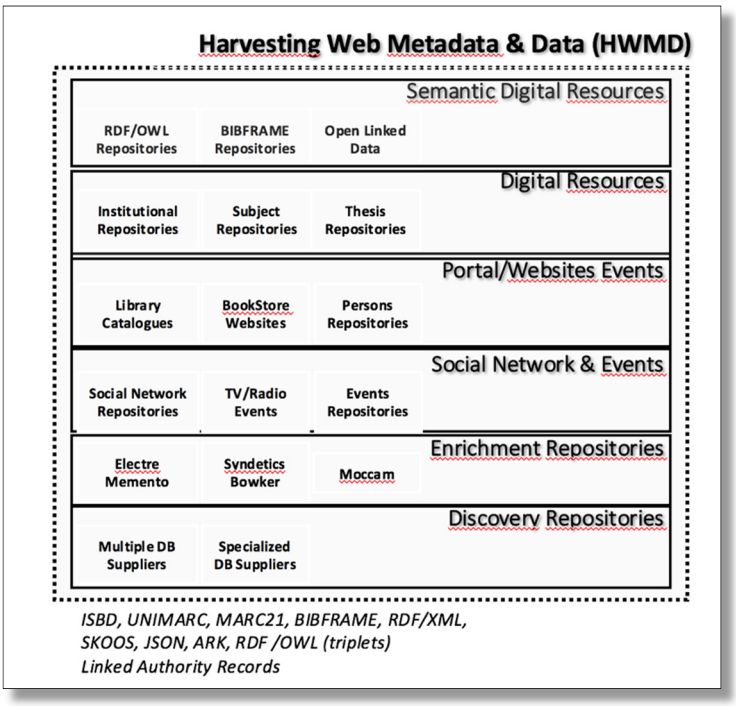


Figure A 1.6 Harvesting of web metadata & data (HWMD)

4.3 Harvesting authority's metadata & data (HAMD)

This sub-section presents the details of the Harvesting of Authority's Metadata & Data (HAMD) as shown in Figure A 1.7.

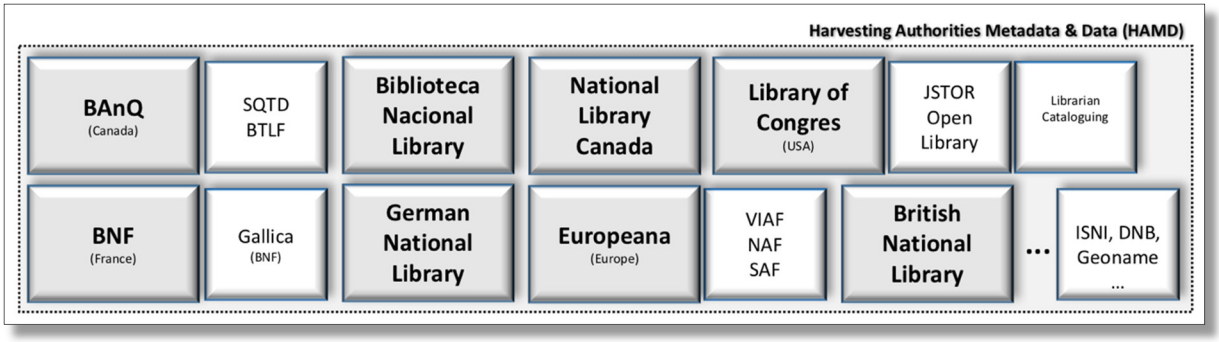


Figure A 1.7 Harvesting of authority's metadata & data (HAMD)

The Semantic Multi-Platform Ecosystem consists of many authority sources, such as:

1. BAnQ (Bibliothèque et Archives nationales du Québec,
2. BAC (Bibliothèque et Archives du Canada,
3. BNF (Bibliothèque Nationale de France),
4. Library of Congress,
5. British Library,
6. Europeana,
7. Spanish Library.

The integration of these platforms in SMESE allows users to build an integrated authority's knowledge base.

4.4 Rules-based semantic metadata external enrichments (RSMEE)

This sub-section presents the details of the rule-based semantic metadata external enrichment engine (RSMEEE), as shown in Figure A 1.8.

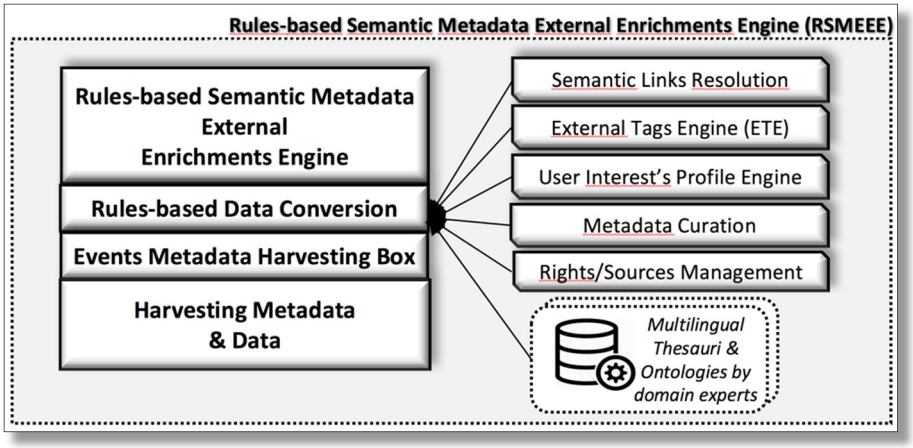


Figure A 1.8 Rules-based semantic metadata external enrichments (RSMEE)

Semantic searches over documents and other content types needs to use semantic metadata enrichment (SME) to find information based not just on the presence of words, but also on their meaning. RSMEEE consists of:

1. Rule-based semantic metadata external enrichment,
2. Multilingual normalization,
3. Rule-based data conversion,
4. Harvesting metadata & data.

Linked open data (LOD) (see Figure A 1.9) based semantic annotation methods are good candidates to enrich the content with disambiguated domain terms and entities (e.g. events, emotions, interests, locations, organizations, persons), described through Unique Resource Identifiers (URIs) (Bontcheva et al., 2015). In addition, the original contents should be enriched with relevant knowledge from the respective LOD resources (e.g. that Justin Trudeau is a Canadian politician). This is needed to answer queries that require common-sense knowledge, which is often not present in the original content. For example: following semantic enrichment, a semantic search for events that provides specific emotions (e.g., happiness, joy) in Montreal according to individual interests this weekend would indeed provide relevant metadata about events in Montreal, even though not explicitly mentioned in the original content metadata.

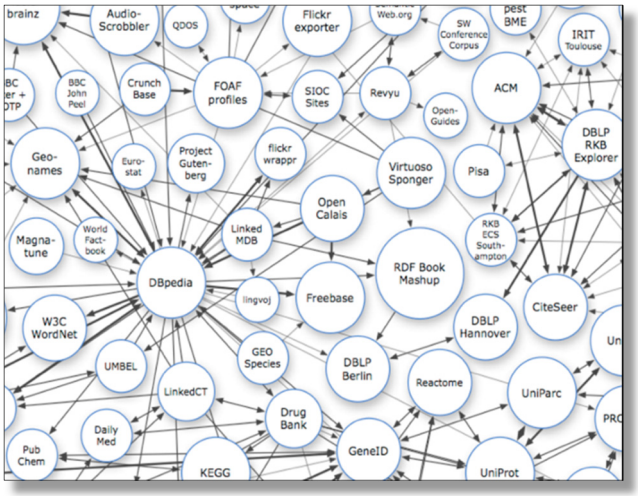


Figure A 1.9 Linked Open Data (LOD)

The semantic annotation process of SMESE creates relationships between semantic models, such as ontologies and persons. It may be characterized as the semantic enrichment of unstructured and semi-structured contents with new knowledge and linking these to relevant domain ontologies/knowledge bases. It typically requires annotating a potentially ambiguous entity mention (e.g. Justin Trudeau) with the canonical identifier of the correct unique entity (e.g. depending on the content - http://dbpedia.org/page/Justin_Trudeau). The benefit of social semantic enrichment is that by surfacing annotated terms derived from the full-text content, concepts buried within the body of the paper/report can be highlighted. Also, the addition of terms affects the relevance ranking in full-text searches. Moreover, users can be more specific by limiting the search criteria to the subject or interest or emotion metadata (e.g. through faceted search).

4.5 Rule-based semantic metadata internal enrichments (RSMIE)

This sub-section presents the details of the rule-based semantic metadata internal enrichment (RSMIE) including software product line engineering (SPLE), as shown in Figure A 1.10.

This sub-system includes:

1. A rule-based semantic metadata internal enrichment,

2. A multilingual normalization process,
3. Software Product Line Engineering (SPLE),
4. A topic, sentiment, emotion, abstract analysis and an automatic literature review.

These processes extract, analyze and catalogue metadata for topics and emotions involved in the SMESE ecosystem. These enrichment processes are based on information retrieval and knowledge extraction approaches. The text is analyzed making use of extension of text mining algorithms such as latent Dirichlet allocation (LDA), latent semantic analysis (LSA), support vector machine (SVM) and k-Means.

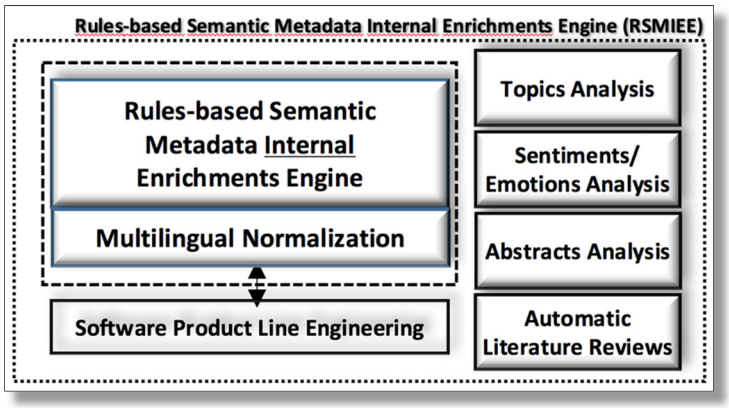


Figure A 1.10 Rule-based semantic metadata internal enrichment (RSMIE)

The different phases of the enrichment process by topics are:

1. Relevant and less similar documents selection phase,
2. Not annotated documents semantic term graph generation phase,
3. Topics detection phase,
4. Training phase,
5. Topics refining phase.

The different phases of the enrichment process by sentiments and emotions are:

1. Sentiment and emotion lexicon generation phase,
2. Sentiment and emotion discovery phase,
3. Sentiment and emotion refining phase.

One of the contributions of the SMESE for digital libraries is that it is not specific to one software product but can be applied to many products dynamically. In addition, it includes a semantic metadata enrichment (SME) process to improve the quality of search and discovery engines.

Indeed, our goal is to provide a SECO that offers a new way to share and learn knowledge. In practice, with the emergence of Big Data, knowledge is not easy to find at the right time and place. The proposed ecosystem uses an SPLE architecture that is a combination of FORM and COPA approaches to catalogue semantically different contents.

Furthermore, we introduce an SPLE decision support process (SPLE-DSP) in order to meet the SPLE characterization such as:

1. Runtime variability functionalities support,
2. Multiple and dynamic binding,
3. Context-awareness and self-adaptation.

SPLE-DSP supports the activation and deactivation of features and changes in the structural variability at runtime and takes into account automatic runtime reconfiguration according to different scenarios. In addition, SPLE-DSP rebinds to new services dynamically based on the description of the relationships and transitions between multiple binding times under an SPLE when the software adapts its system properties to a new context. To take into account context variability to model context-aware properties, SPLE-DSP makes use of an autonomous robot that exploits context information to adapt software behavior to varying conditions.

Furthermore, SPLE-DSP integrates the adaptation of assets and products dynamically. This helps products to evolve autonomously when the environment changes and provides self-adaptive and optimized reconfiguration. Additionally, SPLE-DSP exploits knowledge and context profiling as a learning capability for autonomic product evolution by enhancing self-adaptation.

The SPLE-DSP model is an optimized metadata based reconfiguration model where users select their preferences in terms of configuration of interests.

The dynamic and optimized metadata-based reconfiguration model (DOMRM) takes into account the preferences of several users who have distinct requirements in terms of desirable features and measurable criteria. For example:

1. In terms of hardware criteria, the user can select preferences in terms of memory and power consumption or feature attributes such as internet bandwidth or screen resolution;
2. In terms of software criteria, the user can select the entities and their properties, the property characteristics such as the displaying mode, and expected value type.

Indeed, when user preferences change at runtime, the system must be reconfigured to satisfy as many preferences as possible. Since user preferences may be contradictory, only some will be partially satisfied and a relevant algorithm needed to compute the most suitable reconfiguration. To overcome this drawback, we developed the use of a new metadata-based feature model, referred to as the BiblioMondo semantic feature model (BMSFM), to represent user preferences in terms of semantic features and attributes. Our BMSFM constitutes an evolution of traditional stateful feature models (Trinidad, 2012) that includes the set of user metadata based configurations in the model itself, which allows the representation of user decisions with attributes and cardinalities. More specifically, we developed a metadata-based reconfiguration model that defines all possible metadata and all possible entities that users may need in a specific domain. When a user needs new metadata, he uses the metadata-based request creation tool. The DOMRM model analyses the request and checks whether the requested metadata is relevant and does not already exist. Thus when needed the model automatically creates the new metadata and reconfigures the ecosystem which then becomes available for all users.

Figure A 1.11 illustrates the DOMRM model we designed that is an optimized metadata based configuration for multiple users.

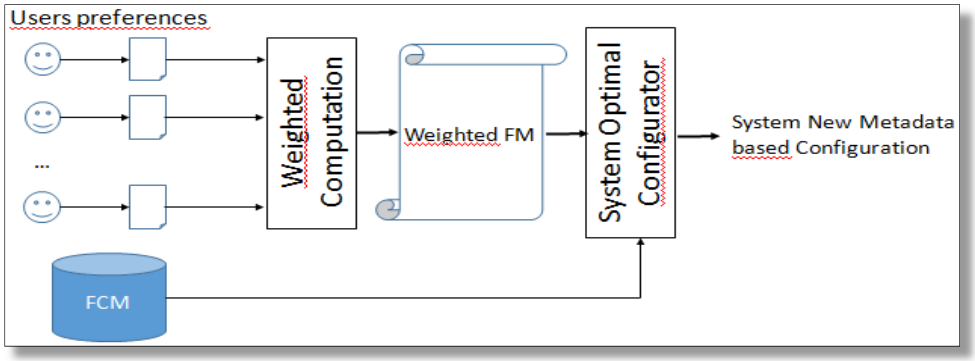


Figure A 1.11 Optimized metadata based configuration for multiple users – DOMRM model

When the user chooses preferences in terms of system behavior, the semantic weight of each feature is computed based on the feature configuration model (FCM). FCM represents the semantic relationship between features where each feature is active or not. In addition, FCM defines the rules that control the activation status of each feature according to its links with the other features. For example, a rule may be: feature Fi should never be activated when Fi-1 is activated. Based on this rule, the model automatically activates or deactivates the feature.

The rules are also used to predict the behavior of the application based on the activation status of features according to user preferences. Notice that each user has his own weight per feature that is defined based on his use of the feature. This weight quantifies the importance of the feature for the user. (More details about the DOMRM algorithm appear in Appendix A).

4.6 Semantic metadata external & internal enrichments synchronization (SMEIES)

This sub-section presents the semantic metadata external & internal enrichment synchronization which represents which processes to synchronize and which enrichments to push outside the ecosystem. See Figure A 1.12.

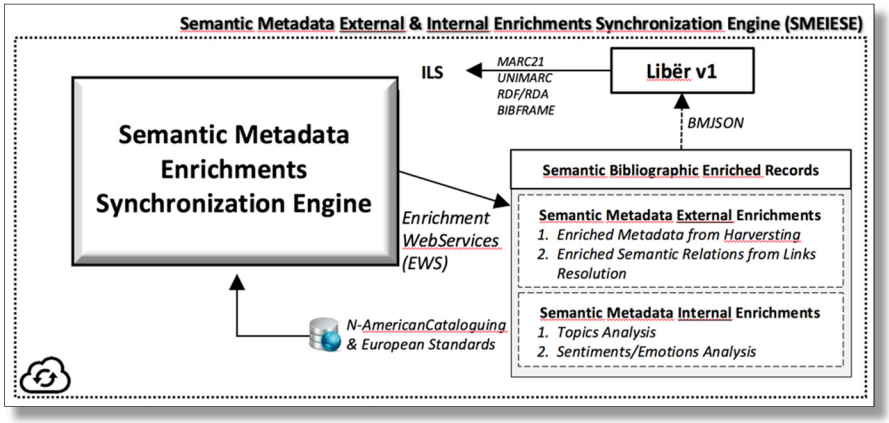


Figure A 1.12 Semantic metadata external & internal enrichment synchronization (SMEIES)

4.7 User interest-based gateway (UIG)

This sub-section presents the user interest-based gateway (UIG) that represents the person (mobile or stationary) who interacts with the ecosystem. See Figure A 1.13.

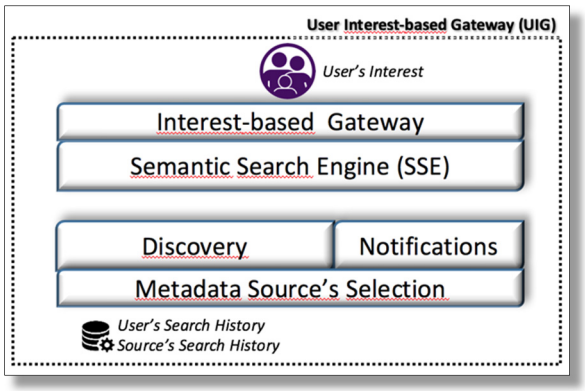


Figure A 1.13 User Interest-based Gateway (UIG)

The users and contributors are categorized into five groups:

1. Interest-based gateway (mobile-first),
2. Semantic Search (SS),
3. Discovery,
4. Notifications,
5. Metadata source selection.

4.8 Semantic master catalogue (SMC)

This sub-section presents the semantic master catalogue (SMC) that represents the knowledge base of the SMESE ecosystem. See Figure A 1.14.

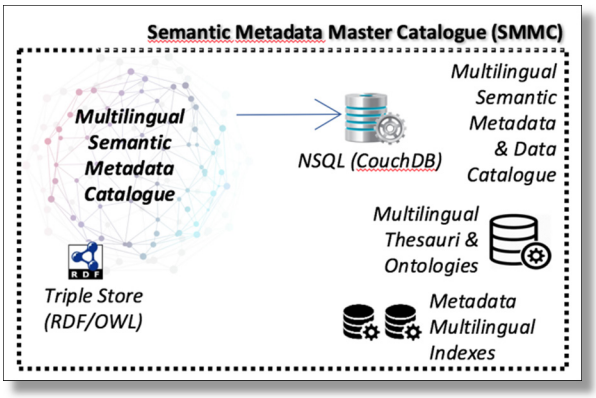


Figure A 1.14 Semantic Master Catalogue (SMC)

4.9 Semantic analytical (SA)

This sub-section presents the semantic analytical (SA) that represents the analytical of the SMESE ecosystem. See Figure A 1.15.

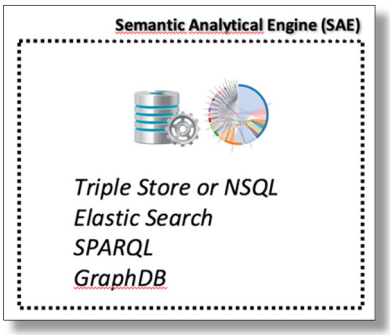


Figure A 1.15 Semantic Analytical (SA)

5. An implementation of SMESE for a large semantic digital library in industry

The proposed SMESE architecture has been implemented for a large digital library. The product InMédia V5 was implemented with a global metadata model defined with all the known entities and constraints. The catalogue contains more than 2 million items, with 18 entities and 132 defined metadata. SMMC identifies 1453 metadata and defines a metamodel that consists of a semantic classification of metadata into meta entities.

In addition to semantic web technologies, the characteristics and challenges of SMESE for large digital libraries are:

1. Automatic cataloguing with the least human intervention,
2. Metadata enrichment,
3. Discovery and definition of semantic relationships between metadata and records,
4. Semi-automatic classification of bibliographic records,
5. Semantic cataloging and validated metadata making use of a multilingual thesaurus.

First, we defined a list of entities, called Meta Entity, which introduced 193 items. These items represent all library materials. In addition, the structure of the model allows addition of new entities as may be required. Figure A 1.16 shows the SMESE meta-entity model where for each entity there is: an ID, propertyName, description, labels in different languages, and the domain that represents the logic group of the entity. The domain may be 'user' as response value for a metadata. In this implementation, all instances of the entities of the domain can be the response value. The ID allows the user to uniquely identify the entity whatever the language, the source of entities or the metadata model (DC, UNIMARC, MARC21, RDA, BIBFRAME).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	META ENTITY														
2						BM label					Client Label			PortFolioEnti	
3	Id.	OldpropertyName	propertyName	Name	Description	Fr	En	Sp	Gr	Ne	Fr	En	Source	tyName	Domain
4	E1-n	Book	book	Book	C'est les contenus du type livre	Livre	Book	"	"	"	Livre	Book	BNF		bibliomondo
5	E2-n	Serial	serial	Serial	C'est les contenus du type périodique	Périodique	Serial	"	"	"	Périodique	Serial			bibliomondo
6	E3-n	Audio	audio	Audio	C'est les contenus du type audio	Audio	Audio	"	"	"	Audio	Audio			bibliomondo
7	E4-n		comic	Comic	C'est les contenus du type bande	Bande	Comic	"	"	"	Bande dessinée	Comic			bibliomondo
8	E5-n		digitalresource	DigitalResour ce	C'est les contenus du type ressource numérique	Ressource Numérique	DigitalResource	"	"	"	Ressource Numérique	DigitalResourc e			bibliomondo
9	E6-n		document	Document	C'est les contenus du type document	Document	Document	"	"	"	Document	Document			bibliomondo
10	E7-n		game	Game	C'est les contenus du type jeu	Jeu	Game	"	"	"	Jeu	Game			bibliomondo
11	E8-n	Image	image	Image	C'est les contenus de type Image	Image	Image	"	"	"	Image	Image	BNF		bibliomondo
12	E9-n	MusicalScore	musicalscore	MusicalScore	C'est les contenus du type partition de musique	Partition de musique	Musical Score	"	"	"	Partition de musique	Musical Score			bibliomondo
13	E10-n	Video	video	Video	C'est les contenus du type vidéo	Vidéo	Video	"	"	"	Vidéo	Video			bibliomondo
14	E11-n	FRBRWork	creativework	Creativework	The most generic kind of creative wo	Oeuvre	FRBRWork	"	"	"	Oeuvre	FRBRWork	BNF		bibliomondo
15	E12-n		manifactation	Manifactatio	C'est la manifestation d'une oeuvre	Manifestation	Manifestation	"	"	"	Manifestation	Manifestation			bibliomondo
16	E13-n		expression	Expression	C'est l'expression d'une	Expression	Expression	"	"	"	Expression	Expression			bibliomondo
17	E14-n		concept	Concept	C'est une concept	Concept	Concept	"	"	"	Concept	Concept			bibliomondo
18	E15-n		city	City	C'est les sous divisons d'un canton	Ville	City	"	"	"	Ville	City	BM		places
19	E16-n		postalAddress	PostalAddres	C'est les contenus de type Adresse	Adresse postal	Postal address	"	"	"	Adresse postal	Postal address	BNF		places
20	E17-n		placeofinterest	PlaceOfInteres st	C'est un lieu qui suscite un intérêt particulié; par exemple, un site touristique, un musé, monument,...	Place d'intérêt	Place of interest	"	"	"	Place d'intérêt	Place of interest	BM		places
21	E18-n		country	Country	C'est un pays	Pays	Country	"	"	"	Pays	Country	BNF		places
22	E19-n		region	Region	C'est les sous divisions d'un pays; c'est le cas des régions en France, des États au États-Unis ou des provinve au	Région	Region	"	"	"	Région	Region	BM		places

Matrix - UsersRights () | Meta Instances | Meta Entity | Meta Metadata | Matrix - Physical (OK) | Matrix - UserProfile (OK) | Matrix - Bibliographie (OK) | Matrix - Journal (OK) | Matri

Figure A 1.16 SMESE Meta Entity model

Next, the list of metadata is defined. 1341 metadata are defined. Each metadata entry has the following additional metadata called Meta Metadata:ID, relatedContentType, is Enrichment, is Repeatable, thesaurus, type, and sourceOfSchema, which are defined as follows:

1. “sourceOfSchema” represents the origin of the metadata;
2. “id” allows unique identification of the entity;
3. “propertyName” is a comprehensive term that defines this metadata;
4. “UNIMARC”, “MARC21”, “propertyName” allow users to create a mapping between them to make them interoperable;
5. “UNIMARC” and “MARC21” are codes such as 300$abcf;
6. “Expected type” represents the type of value that may be assigned to the metadata as response;
7. “isRelated” denotes that the response of the metadata is an entity where the identity is given by “relatedContentType”;
8. “thesaurus” mentions the thesaurus name that is used to control the responses to assign to the metadata;

9. “type” allows classification of the metadata as “descriptive”, “structural”, “administrative”, “dimension”, “longevity” or “identification”.

This classification allows users to do meta research. Figure A 1.17 shows an illustration of the Meta Metadata model.

A2 | Id.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
1								META METADATA																			
2								BM label					Client Label														
3	Id.	Unimarc	Marc21	OldpropertyName	propertyName(BanQ)	propertyName(Excel)	Description	Fr	En	Sp	Ne	Gr	Fr	En	Expected Type	isRelated	relatedContentType	Short Notice	Enriched	Data Location	Hidden	Repeatable	Thesaurus	Type	isUrl	Source OfSchema	dia Version
4	M1-m	200$aodehi	OP0245	OP0245	title	title	It is the main title of a content	Titre	Title				Titre	Title	Text									Desc		BM	
5	M2-m	700$abcdfg4 / 710$abcdefgh 4 / 720$af4	OP01XX	OP01XX	author	author	The author of this content. Please note that author is special in that HTML 5 provides a special mechanism for indicating authorship via the rel tag. That	Auteur	Author				Auteur	Author	Text	X	domain:personality	X		Local			Personnes ou Organisations	Desc		BNF	
6	M3-m	701$abcdfg4 / 711$abcdefgh 4 / 721$af4	OP01XX	OP01XX	creator	creator	The creator/author of this CreativeWork. This is the same as the Author property for CreativeWork	Auteur	Author				Composeur/ Interprète	Composer/ Performer	Text	X	domain:personality	X		Local			Personnes ou Organisations	Desc		BNF	
7	M4-m		OP001XX	OP001XX		promoter	C'est le organisateur du contenu; c'est le cas des évènements	Organisateur	Promoter				"	"	Text	X	domain:personality			Local			Personnes ou Organisations	Desc		BM	
8	M5-m	215$aode	OP0300	OP0300	dcterms:descri	description	A short description of the item.	Description	Description				Description	Description	Text					Local		X		Desc		BNF	
9	M6-m				Image	Image	An image of the item. This can be a image file or a URL or a fully described	Imagette	Thumbnail				Imagette	Thumbnail	Image				X	BM.com		X		Desc		BNF	
10	M7-m	101$a	LO0546	LO0546	dcterms:language	language	C'est la langue (parlée ou écrite) dans laquelle le contenu est créé ou dans laquelle une personne s'exprime; par exemple, un livre rédigé en français aura	Langue(s)	Language(s)				Langue(s)	Langue(s)	list[FACET.langue_s]					Local		X	Langues	Desc		BM	
11	M8-m	540$ae	OP0740	OP0740	titleAddedUncontrolledRelat	titleAddedUncontrolledRelated	C'est d'autres titres du contenu qui pourraient être utilisés dans d'autres	Autres titres	Other titles				Autres titres	Other titles	Text					Local				Desc		BM	
12	M9-m	500$abhijklmnqrsuvw	OP0240	OP0240	titleUniform	titleUniform	C'est le titre unifié du contenu	Titre uniforme	Uniform title				Titre uniforme	Uniform title	Text					Local				Desc		BM	
13	M10-m					alsoAvailableIn	C'est les autres langues dans lesquelles le contenu a été aussi produit ou dans lesquelles une personne s'exprime	Disponible en d'autres langues	Available in other languages				Aussi disponibe en	Also available in	list[FACET.langue_s]					Local		X	Langues	Desc		HEC OIC	
14	M11-m	330$a	LO0520	LO0520	abstract	abstract	C'est une présentation abrégée qui rend compte de l'essentiel en ce qui concerne	Résumé	Abstract				Résumé	Summary	Text					Local				Desc		BM	
15	M12-m					abstractInMedia	C'est le résumé obtenu à partir d'une analyse du contenu	Résumé InMédia	InMedia Summary				"	"	Text				X	BM.com				Desc		BM	
16	M13-m	702$abcdfg4 / 712$abcdefgh 4 / 716$acf / 722$af4 /	OP07XX	OP07XX	contributor	contributor	C'est les autres personnes qui ont contribuées à la réalisation du contenu	Contributeurs	Contributors				Composeurs/ Auteurs/ Interprètes	Composers/ Songwriters/ Singers	Text	X	domain:personality			Local			Personnes ou Organisations	Desc		BM	
17	M14-m		AF0508	AF0508	credit	credit	C'est les contributions au contenu	Contributio	Credits				Contributio	Credits	Text					Local				Desc		BM	
18	M15-m				rdarelworkMa	workManifeste	C'est l'oeuvre qui a permie de créer le	Deuvre	Work				Selon	Work	Text	X	creativeW			Local				Desc		BM	

Matrix - UsersRights () | Meta Instances | Meta Entity | Meta Metadata | Matrix - Physical (OK) | Matrix - UserProfile (OK) | Matrix - Bibliographie (OK) | Matrix - Journal (OK) | Matri

Figure A 1.17 SMESE metadata model

The semantic matrix model is defined for each entity based on the metaentity and metadata model. This semantic matrix model allows users to define a metadata matrix for each entity where a metadata matrix denotes the logical subset of metadata of metadata model that describes a given entity. Figure A 1.18 illustrates an example of a semantic metadata matrix for a specific content. The objective behind the matrix is to allow the reuse of metadata for distinct entities. This extends the search range for entities, facilitates the search for users in terms of search criteria and increases the probability of achieving satisfying results.

A5 ='Meta Metadata'!A841

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
1	SEMANTIC MATRIX						Physical													E90-n							
2							META METADATA																				
3								BM label					Client Label														
4	Id.	Unimarc	Marc21	OldpropertyNam	propertyName (Bau0)	propertyName (Excel)	Description	Fr	En	Sp	Ne	Gr	Fr	En	Expected Type	isRelated	relatedContentTy	Short Notic	Earched	Data Location	Hidden	Repeatabl	Thesaurus	Type	isUrl	Source	Version InMedia
5	M854-m	0	0	0	0	bmsin	Identification Number; all the content has this metadata	BMSIN	BMSIN	0	0	0	BMSIN	BMSIN	Text	0	0	0	0	BM.com	0	0	0	0	0	BM	V5.2
6	M855-m	0	0	0	0	dateOfcataloging	It is the date of cataloging	Date de catalogage	Cataloging date	0	0	0	Date de catalogage	Cataloging date	Date	0	0	0	0	BM.com	0	0	0	0	0	BM	V5.2
7	M868-m	0	0	0	0	isni	International Standard Name Identifier	ISNI	ISNI	0	0	0	ISNI	ISNI	Text	0	0	0	0	BM.com	0	0	0	0	0	BM	V5.2
8	M3-m	g4 / 111$abcdef	OP01XX	OP01XX	creator	creator	CreativeWork. This is the same as the Author property for CreativeWork	Auteur	Author	0	0	0	Compositeur/ Interprète	Composer/ Performer	Text	X	domainpersonality	X	0	Local	0	0	es ou Organis	Desc	0	BNF	V5.1
9	M215-m	0	OP07XX	OP07XX	contributor	contributorFormatted	A secondary contributor to content	Collaborateurs	Contributors	0	0	0	"	"	Text	X	domainpersonality	0	0	Local	0	0	0	0	0	BM	V5.1
10	M38-m	610$a	OP06XX	OP06XX	subject	subject	It is the subject of a content	Sujets	Subjects	0	0	0	Sujets	Subjects	list[Subject]	0	0	0	0	Local	0	X	Vedette Matiere	Admin	0	BM	V5.1
11	M321-m	0	0	0	documentType	documentType	C'est le type de document du contenu	Type de document	Document type	0	0	0	Type de document	Document type	list[typeDocument]	0	0	0	0	Local	0	0	0	0	0	BM	V5.1
12	M322-m	0	0	0	documentTypeCluster	documentTypeCluster	C'est le groupe de type de document du contenu	Type de cluster de document	Cluster type of document	0	0	0	Type de cluster de	Cluster type of	Text	0	0	0	0	Local	0	0	0	0	0	BM	V5.1
13	M323-m	001	0	0	identifierBnf	identifierDataBnf	C'est l'identifiant bnf du contenu	Identifiant Bnf	Identifier Bnf	0	0	0	identifierBnf	identifierBnf	Text	0	0	0	0	Local	0	0	0	0	0	BM	V5.1
14	M131-m	0	0	0	dateIssuedFirst	dateIssuedFirst	C'est la date de la première édition du contenu	Date de la première édition	Date of first edition	0	0	0	Première édition	First edition	Text	0	0	0	0	Local	0	0	0	Dim	0	BM	V5.1
15	M130-	0	0	0	dateIssued	dateIssued	C'est la date d'édition du contenu	Date Édition	Edition date	0	0	0	"	"	Date	0	0	0	0	Local	0	0	0	Dim	0	BM	V5.1
16	M133-m	103	0	0	rdvgfdateOfPublicationManifestati	dateIssuedFormatted	Date of first broadcast/publication	Date de publication	Date of publication	0	0	0	Date de publication	Date of publication	Date	0	0	0	0	Local	0	0	0	Dim	0	BM	V5.1
17	M1-m	200$acdeh	OP0245	OP0245	title	title	It is the main title of a content	Titre	Title	0	0	0	Titre	Title	Text	0	0	0	0	0	0	0	0	Desc	0	BM	V5.1
18	M23-m	0	OP05XX	OP05XX	note	note	C'est les notes associées au contenu	Notes	Notes	0	0	0	Notes	Notes	list [Note]	0	0	0	0	Local	0	0	0	Desc	0	BM	V5.1
19	M37-m	010$a	OP0020	OP0020	isbn	isbn	International Standard Book Number; il est possible de faire des entrées	ISBN	ISBN	0	0	0	ISBN	ISBN	Text	0	0	0	0	Local	0	X	0	Desc	0	BM	V5.1
20	M36-m	073$a	AM0024	AM0024	ean	ean	European Article Number; il est possible de faire des entrées	EAN	EAN	0	0	0	EAN	EAN	Text	0	0	0	0	Local	0	X	0	Desc	0	BM	V5.1
21	M38-m	011$a	OP0022	OP0022	issn	issn	International Standard Serial Number; il est possible de faire des entrées	ISSN	ISSN	0	0	0	ISSN	ISSN	Text	0	0	0	0	Local	0	X	0	Desc	0	BM	V5.1
22	M218-m	071$ab	OP0028	OP0028	publisherNumber	publisherNumber	C'est le numéro d'un éditeur	No d'éditeur	Publisher no.	0	0	0	No d'éditeur	Editor no.	list text	0	0	0	0	Local	0	0	0	0	0	BM	V5.1
23	M40-m	205$a|g	OP0250	OP0250	edition	edition	C'est l'édition du contenu	Édition	Edition	0	0	0	Édition	Edition	Text	0	0	0	0	Local	0	0	0	Desc	0	BM	V5.1
24	M324-	0	0	0	holdingHas	holdingHas	C'est le Holding Has du contenu	holdingHas	holdingHas	0	0	0	holdingHas	holdingHas	Text	0	0	0	0	Local	0	0	0	0	0	BM	V5.1
25	M325-	0	0	0	holdingId	holdingId	C'est le Holding ID du contenu	holdingId	holdingId	0	0	0	holdingId	holdingId	Text	0	0	0	0	Local	0	0	0	0	0	BM	V5.1
26	M326-m	0	0	0	holdingMedia	holdingMedia	C'est le Holding Media du contenu	holdingMedia	holdingMedia	0	0	0	holdingMedia	holdingMedia	Text	0	0	0	0	Local	0	0	0	0	0	BM	V5.1
27	M123-m	0	0	0	rdvgfplaceOfPublication	placeOfPublication	C'est le lieu de publication du contenu	Lieu de publication	Place of publication	0	0	0	Lieu de publication	Place of publication	Text	X	domainplaces	0	0	Local	0	0	0	Dim	0	BM	V5.1
28	M219-m	210$c	0	0	rdvgfpublishersName	publishersName	C'est le nom d'un éditeur	Nom d'éditeur	Publisher's name	0	0	0	Nom d'éditeur	Publisher's name	0	0	0	0	0	Local	0	0	0	0	0	BM	V5.1
29	M16-m	0	0	0	rdvarchexpressionManifested	expressionManifested	C'est l'expression qui a permie de créer le contenu	Expression	Expression	0	0	0	Expression	Expression	Text	X	expression	0	0	Local	0	0	0	Desc	0	BM	V5.1
30	M15-m	0	0	0	rdvarchworkManifested	workManifested	C'est l'oeuvre qui a permie de créer le contenu	Oeuvre	Work	0	0	0	Sélon l'oeuvre	Work related	Text	X	creativeWork	0	0	Local	0	0	0	Desc	0	BM	V5.1
31	M20-m	0	0	0	rdfsseeAlso	seeAlso	C'est l'apergu du contenu dans le catalogue de la BNF	Catalogue de la BNF	BNF catalog	0	0	0	Catalogue de la BNF	BNF catalog	URL	0	0	0	X	BM.com	0	0	0	Desc	X	BM	V5.1

Matrix - UsersRights () | Meta Instances | Meta Entity | Meta Metadata | Matrix - Physical (OK) | Matrix - UserProfile (OK) | Matrix - Bibliographie (OK) | Matrix - Journal (OK) | Matrix

Figure A 1.18 Example of a SMESE semantic matrix model

After the definition of entities of collections and harvesting of metadata from the dispersed collections, a metadata crosswalk is carried out. This is a process in which relationships among the schema are specified, and a unified schema is developed for the selected collection. It is one of the important tasks for building “semantic interoperability” among collections and making the new digital library meaningful.

The most frequent issues regarding mapping and crosswalks are: incorrect mappings, misuse of metadata elements, confusion in descriptive metadata and administrative metadata, and lost information. Indeed, due to the varying degrees of depth and complexity, the crosswalks among metadata schemas may not - necessarily be equally interchangeable. To solve the issue of varying degrees of depth, we developed atomic metadata: these metadata allow description of the most elementary aspects of an entity. It then becomes easy to map all metadata from any schema.

Figure A 1.19 illustrates a mapping ontology model where relationships are in red while simple descriptions are in black.

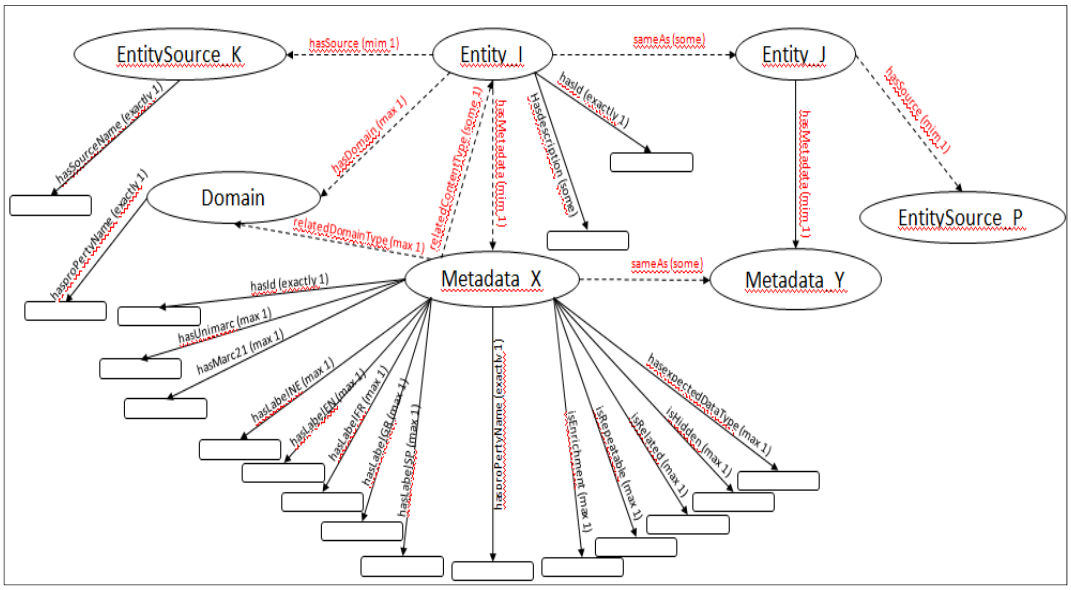


Figure A 1.19 Ontology mapping model

Figure A 1.20 shows that each entity has at a minimum one source of schema denoted by the relationship “hasSource” and a minimum of one metadata denoted by the relationship “hasMetadata”. The relationship “sameAs” is used to denote the mapping between distinct metadata or entity schema source.

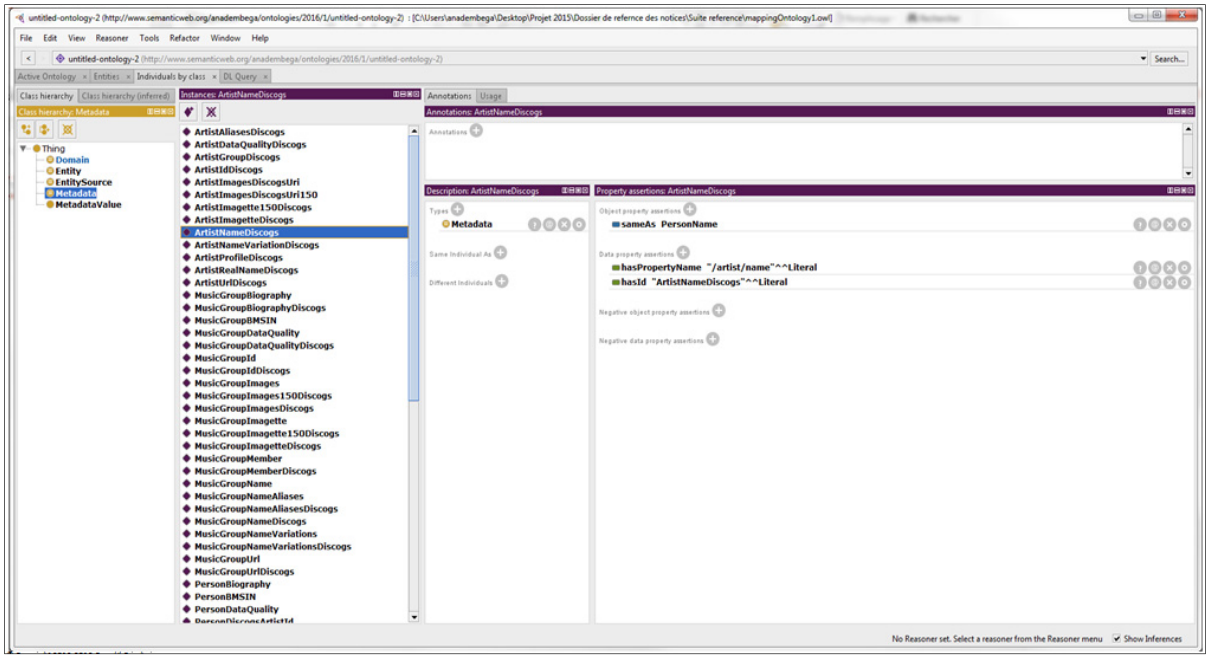


Figure A 1.20 Ontology mapping implementation using Protégé

The output of the ontology is an OWL file. This OWL file is used by a crosswalk to automatically assign metadata value that are harvested from distinct sources. In the proposed ecosystem two sources are harvested: Discogs (www.discogs.com) for music and ResearchGate (www.researchgate.net) for academic papers.

A total of 94,015,090 metadata records were collected from these two sources:

- From Discogs, we collected 7,983,288 entities: 2,621,435 music releases, 4,466,660 artists and 895,193 labels;
- From researchGate, we collected 86,031,802 entities: 77,031,802 publications and more than 9,000,000 researchers.

In fact, SMESE contains more than 3.4 billions triplets and growing.

6. Summary and future work

In this paper, we proposed a design and implementation of a semantic enriched metadata software ecosystem (SMESE).

The SMESE prototype, which was implemented at BiblioMondo, integrates data and metadata enrichment to support specific applications for distributed content management. To perform this integration, SMESE makes use of the software product line engineering (SPLE) approach, a component-based software development (CBSD) approach and our proposed new concept, called semantic metadata enrichment (SME) with distributed contents and mobile first design (MFD). In this implementation, the SPLE architecture is a combination of FORM and COPA approaches.

We also presented our implementation of SMESE for digital libraries. This included SPLE-DSP, a new decision support process for SPLE. SPLE-DSP consists of a dynamic and optimized metadata based reconfiguration model (DOMRM) where users select their preferences in the market place. SPLE-DSP takes into account runtime variability functionalities, multiple and dynamic binding, context-awareness and self-adaptation.

We also implemented the Meta Entity that represents all library materials and meta metadata. The ontology mapping model was then implemented to make our models interoperable with existing metadata models such as Dublin Core, UNIMARC, MARC21, RDF/RDA and BIBFRAME.

The major contributions of this paper are as follows:

1. Definition of a software ecosystem architecture (SMESE) that configures the application production process including software aspects based on CBSD and SPLE approaches;
    a. The use of a LOD-based semantic enrichment model for semantic annotation processes;
    b. The integration of National Research Council of Canada (NRC) emotion lexicon for emotion detection;
    c. A repository of 43 thesaurus included in RAMEAU for semantical contextualization of concepts;
    d. An extended latent Dirichlet allocation (LDA) algorithm for topic modelling;

2. Definition and partial implementation of semantic metadata enrichment using metadata SPLE and an SMMC (semantic master metadata catalogue) to create a universal metadata knowledge gateway (UMKG);
3. The design and implementation of an SMESE prototype of for a semantic digital library (Libër).

This paper proposed a semantic metadata enrichments software ecosystem (SMESE) to support multi-platform metadata driven applications, such as a semantic digital library. Our SMESE integrates data and metadata based on mapping ontologies in order to enrich them and create a semantic master metadata catalogue (SMMC).

Within the SPLE context, SPLE-DSP is used by SMESE to support dynamic reconfiguration. This consists of a dynamic and optimized metadata based reconfiguration model (DOMRM) where users select their preferences within the market place. SPLE-DSP takes into account runtime metadata-based variability functionalities, multiple and dynamic binding, context-awareness and self-adaptation. Our SMESE represents more than 200 million relationships (triplets). Figure A 1.21 represents, in blue, the - implemented SMESE platform.

Future work will include:

1. An enhanced ecosystem and rule-based algorithms to enrich metadata semantically, including topics, sentiments and emotions;
2. Evaluation of the performance of an implementation of the SMESE ecosystem using different projects, comparing- results against existing techniques of metadata enrichments;
3. Exploring text summarization and automatic literature review as metadata enrichment. The semantic annotations could be used to enrich metadata and provide new types of visualizations by chaining documents backward and forward inside automated literature reviews.

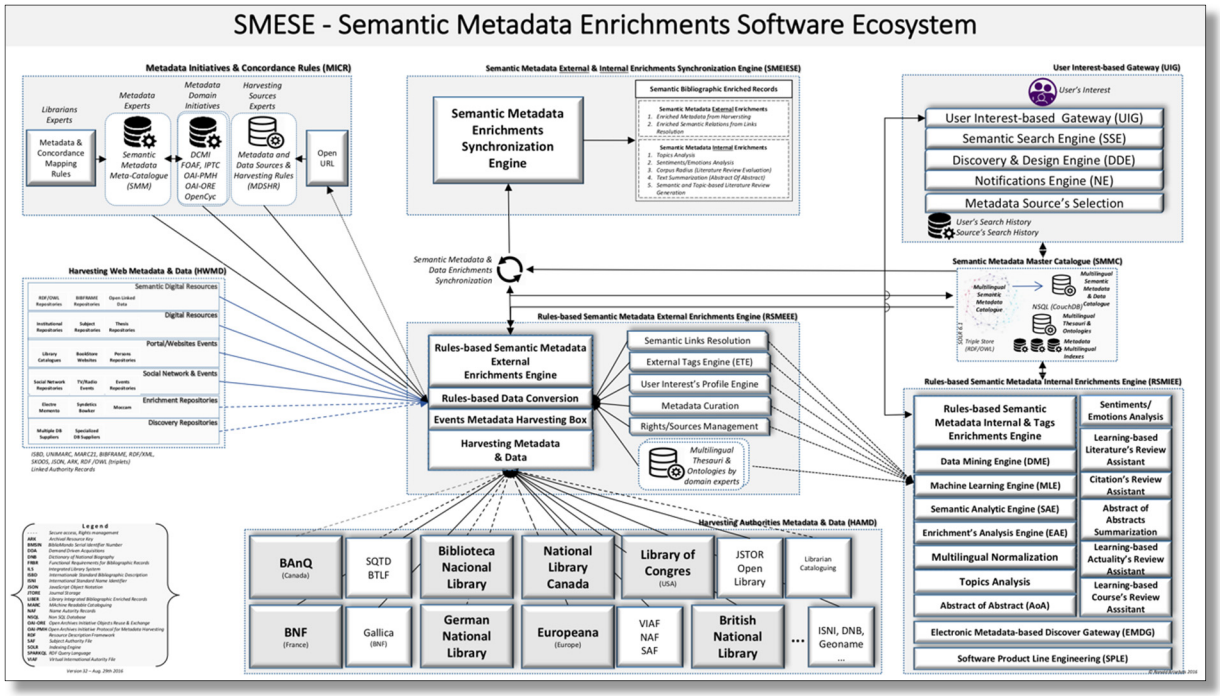


Figure A 1.21 Proposed SMESE architecture: semantic enriched metadata software ecosystem

Appendix A: Dynamic and Optimized Metadata-based Reconfiguration Model (DOMRM)

This Appendix presents the details of the DOMRM model. The main idea behind DOMRM is the more a user uses a specific feature, the more his weight for this feature increases. The weight UjFi of user j for feature i is given by:

$$UjFi = \frac{n(Uj, Fi)}{\sum_{k=1}^{P} n(Uk, Fi)} \quad \text{(A 1.1)}$$

where n(Uj,Fi) denotes the number of times user j used the feature i.

Making use of user weight per feature and their preferences, the feature weight that determines its activation or not is computed. Considering that US is the set of users who have selected a feature Fi (activation of feature), and UR is the set of users who have removed that feature (deactivation of feature), the value 1 is assigned when a user actives the feature, and -1 when

he removes it. Let c(Uj,Fi) be the choice of user j for the activation status of feature Fi. The weight of feature Fi can be defined -using the following formula:

$$w(Fi) = \begin{cases} 1 & whether\ 0 < \sum_{Uk \in US \cup UR} [c(Uk,Fi) \times UkFi] \\ -1 & whether\ 0 > \sum_{Uk \in US \cup UR} [c(Uk,Fi) \times UkFi] \end{cases} \quad \text{(A 1.2)}$$

The computed weight of each feature allows one to define the weight FM that is used by the system optimal configurator with the FCM to generate the new configuration of the system for all users. When the feature weight is negative and the FIS rules allow de-activation, the feature is deactivated and when the feature weight is positive and the FIS rules allow activation the DOMRM model activates the feature. The activation status of the feature is not modified when the feature weight is null and the current activation status is conserved.

APPENDIX II

A Semantic Metadata Software Ecosystem based on Topic and Sentiment/Emotion Analysis Enrichment (SMESE V3)

Ronald Brisebois[1], Alain Abran[2], Apollinaire Nadembega[1], Philippe N'techobo[1]

[1] Bibliomondo, Montréal, Canada
{ronald.brisebois,apollinaire.nadembega,philippe.ntechobo}@bibliomondo.com
[2] École de technologie supérieure, Université du Québec, Canada,
alain.abran@etsmtl.ca

Paper submitted for publication to Information Systems, November 2016

Abstract

Semantic information retrieval is frequently used to extract meaningful information from the unstructured web and from long texts. As existing computer search engines struggle to understand the meaning of natural language, semantically enriching entities with meaningful metadata may improve search engine capabilities.

In a previous paper, SMESE for semantic metadata enrichment software ecosystem based on a multi-platform metadata model has been proposed. This paper presents an enhanced version with interest-based enrichments named SMESE V3.

This paper proposes to help users finding interest-based contents, through text analysis approaches for sentiments and emotions detection. SMESE V3 can be used (or: makes it possible) to create a semantic master catalogue with enriched metadata that enables interest-based search and discovery. This paper presents the design, implementation and evaluation of a SMESE V3 platform using metadata and data from the web, linked open data, harvesting and

concordance rules, and bibliographic record authorities. It includes three distinct processes that:

1. Discover enriched sentiment and emotion metadata hidden within the text or linked to multimedia structure using the proposed BM-SSEA (BM-Semantic Sentiment and Emotion Analysis) algorithm;
2. Implement rule-based semantic metadata internal enrichment (RSMIEE includes algorithms BM-SATD and BM-SSEA);
3. Generate semantic topics by text, and multimedia content analysis using the proposed BM-SATD (BM-Scalable Annotation-based Topic Detection) algorithm.

The performance of the proposed ecosystem is evaluated using a number of prototype simulations by comparing them to existing enriched metadata techniques. The results show that the enhanced SMESE V3 and its algorithms enable greater understanding of content for purposes of interest-based search and discovery.

Keywords: emotion detection, natural language processing, semantic topic detection, semantic metadata enrichment, sentiment analysis, text and data mining.

1. Introduction

The rapid development of search and discovery engines, the sudden availability of millions of documents, and the millions upon millions of relationships to linked documents from a growing multitude of sources (e.g., online media, social media and published documents) all make it challenging for a user to find documents relevant to his or her interests or emotions.

Currently, rich information within text data can be utilized to reveal some meaningful semantic metadata, such as sentiments, emotions, and semantic relationships. Semantic information retrieval (SIR) is the science of searching semantically for information within databases, documents, texts, multimedia files, catalogues and the web.

The human brain has an inherent ability to detect topics, emotions, relationships or sentiments in written or spoken language. However, the internet, social media and repositories have expanded the number of sources, volume of information and number of relationships so fast that it has become difficult to process all this information (Appel et al., 2016).

The goal is to increase the findability of entities matching user interest using external (outside documents) and internal (within documents) semantic metadata enrichment algorithms. While computer search engines struggle to understand the meaning of natural language, semantically enriching entities with meaningful metadata may improve those capabilities. Words themselves are often used inconsistently, having a wide variety of definitions and interpretations. Although there may be no relationship between individual words of a topic, sentiment or emotion, thesauri do express associative relationships between words, ontologies, entities and a multitude of relationships represented as triplets.

Finding bibliographic references or semantic relationships in texts makes it possible to localize specific text segments using ontologies to enrich a set of semantic metadata related to topics, sentiments and emotions. This paper presents an enhanced implementation of SMESE using metadata and data from linked open data, structured data, metadata initiatives, concordance rules and authority's metadata to create a semantic metadata master catalogue.

The current methodology proposed by SIR researchers for text analysis within the context of entity metadata enrichment (EME) reduces each document in the corpus to a vector of real numbers where each vector represents ratios of counts. Several EME approaches have been proposed, most of them making use of term frequency–inverse document frequency (tf-idf) (Niu et al., 2016; Salton & Buckley, 1988). In the tf-idf scheme, a basic vocabulary of "words" or "terms" is chosen, then for each document in the corpus, a frequency count is calculated from the number of occurrences of each word (Niu et al., 2016; Salton & Buckley, 1988). After suitable normalization, the frequency count is compared to an inverse document frequency count (e.g the inverse of the number of documents in the entire corpus where a given word occurs — generally on a log scale, and again suitably normalized). The end result is a term-by-document matrix X whose columns contain the tf-idf values for each of the documents in the corpus. Thus the tf-idf scheme reduces documents of arbitrary length to fixed-length lists

of numbers. For non-textual content, tools are available to extract the text from multimedia entities. For example, Bougiatiotis and Giannakopoulos (Bougiatiotis & Giannakopoulos, 2016) propose an approach that extracts topical representations of movies based on mining of subtitles. This paper focuses on contributions to mainly one EME research fields: sentiment analysis (SA).

The main objective of sentiment analysis (SA) is to establish the attitude of a given person with regard to sentences, paragraphs, chapters or documents (Appel et al., 2016; Balazs & Velásquez, 2016; Kiritchenko, Zhu, & Mohammad, 2014; Niu et al., 2016; Patel & Madia, 2016; Ravi & Ravi, 2015; Serrano-Guerrero et al., 2015; Taboada, Brooke, Tofiloski, Voll, & Stede, 2011; Vilares, Alonso, & GÓMez-RodrÍGuez, 2015). Indeed, many websites offer reviews of items like books, cars, mobiles, movies etc., where products are described in some detail and evaluated as good/bad, preferred/not preferred; unfortunately, these evaluations are insufficient for users in order to help them to make decision. In addition, with the rapid spread of social media, it has become necessary to categorize these reviews in an automated way (Niu et al., 2016).

For this automatic classification, there are different methods to perform SA, such as keyword spotting, lexical affinity and statistical methods. However, the most commonly applied techniques to address the SA problem belong either to the category of text classification supervised machine learning (SML), which uses methods like naïve Bayes, maximum entropy or support vector machine (SVM), or to the category of text classification unsupervised machine learning (UML).

Also, fuzzy sets appear to be well-equipped to model sentiment-related problems given their mathematical properties and ability to deal with vagueness and uncertainty —characteristics that are present in natural languages processing.

Thus, a combination of techniques may be successful in addressing SA challenges by exploiting the best of each technique. In addition, the semantic web may be a good solution for searching relevant information from a huge repository of unstructured web data (Patel & Madia, 2016).

According to (Balazs & Velásquez, 2016), the SA process typically consists of a series of steps:

1. Corpus or data acquisition,
2. Text preprocessing,
3. Opinion mining core process,
4. Aggregation and summarization of results,
5. Visualization.

One current limitation in the area of SA research is its focus on sentiment classification while ignoring the detection of emotions. For example, document emotion analysis may help to determine an emotional barometer and give the reader a clear indication of excitement, fear, anxiety, irritability, depression, anger and other such emotions. For this reason, we focus on sentiment and emotion analysis (SEA) instead of SA.

A number of algorithms or approaches are used to perform text mining, including: latent Dirichlet allocation (LDA) (David M. Blei et al., 2003), tf-idf (Niu et al., 2016; Salton & Buckley, 1988), latent semantic analysis (LSA) (Dumais, 2004), formal concept analysis (FCA) (Cigarrán et al., 2016), latent tree model (LTM) (P. Chen et al., 2016), naïve Bayes (NB) (Moraes et al., 2013), support vector machine method (SVM) (Moraes et al., 2013), artificial neural network (ANN) (Ghiassi et al., 2013) based on the associated document's features.

Our approach improves the accuracy of topic detection, sentiment and emotion discovery by semantically enriching the metadata from the linked open data and the bibliographic records existing in different formats. This paper presents the design, implementation and evaluation of an enhanced ecosystem, called semantic metadata enrichment ecosystem or SMESE V3. It includes:

1. An enhanced semantic metadata meta-catalogue,
2. An enhanced harvesting of metadata & data,
3. Metadata enrichment based on semantic topic detection, sentiment and emotion analysis.

More specifically, SMESE V3 consists of processes implementing two rule-based algorithms to enrich metadata semantically:

1. BM-SATD: generation of semantic topics by text analysis, relationships and multimedia contents;
2. BM-SSEA: discovery of sentiments and emotions hidden within the text or linked to a multimedia structure through an AI computational approach.

Using simulation, the performance of SMESE V3 was evaluated in terms of accuracy of topic detection, sentiment and emotion discovery. Existing approaches to enriching metadata (e.g., topic detection, sentiment or emotion discovery) were used for comparison. Simulation results showed that SMESE V3 outperforms existing approaches.

The remainder of the paper is organized as follows. Section 2 presents the related work. Section 3 describes SMESE V3 and its various algorithms while Section 4 presents the prototype of the SMESE V3 multiplatform architecture developed. Section 5 presents the evaluation through a number of simulations. Section 6 presents a summary and some suggestions for future work.

2. Related work

In the past few years, a number of natural language processing (NLP) tasks have been configured for semantic web (SW) tasks including: ontology learning, linked open data, entity resolution, natural language querying to linked data, etc. (Gangemi, 2013). This improvement of metadata enrichment using SW involves obtaining hidden data, hence the concept of entity metadata extraction (EME).

Interest in EME was initially limited to those in the SW community who preferred to concentrate on manual design of ontologies as a measure of quality. Following linked data bootstrapping provided by DBpedia, many changes ensued with a consequent need for substantial population of knowledge bases, schema induction from data, natural language access to structured data, and in general all applications that make for joint exploitation of structured and unstructured content. In practice, NLP research started using SW resources as

background knowledge. Graph-based methods, meanwhile, were incrementally entering the toolbox of semantic technologies at large.

In the related work section, two fields of entity metadata extraction research from text aspect are investigated:

1. Sentiment and emotion analysis (SEA),
2. Semantic topic detection (STD), see Appendix C – Semantic topic detection.

2.1 Sentiment and emotion analysis

2.1.1 Sentiment analysis

The problem of sentiment analysis has been widely studied and different approaches applied, such as machine learning (ML), natural language processing (NLP) and semantic information retrieval (SIR).

There are three main techniques for sentiment analysis (Shivhare & Khethawat, 2012):

1. Keyword spotting,
2. Lexical affinity,
3. Statistical methods.

Keyword spotting includes developing a list of keywords that relate to a certain sentiment. These words are usually positive or negative adjectives since such words can be strong indicators of sentiment. Keyword spotting classifies text by affect categories based on the presence of unambiguous affect words such as happy, sad, afraid, and bored.

Lexical affinity is slightly more sophisticated than keyword spotting. Rather than simply detecting obvious affect words, it assigns to arbitrary words a probabilistic 'affinity' for a particular emotion. Lexical affinity determines the polarity of each word using different unsupervised techniques. Next it aggregates the word scores to obtain the polarity score of the text. For example, 'accident' might be assigned a 75% probability of indicating a negative effect, as in 'car accident' or 'injured in an accident'.

Statistical methods, such as Bayesian inference and support vector machines, are supervised approaches in which a labeled corpus is used for training a classification method which builds a classification model used for predicting the polarity of novel texts. By feeding a large training corpus of affectively annotated texts to a machine learning algorithm, it is possible for the system to not only learn the affective valence of affect keywords (as in the keyword spotting approach), but also to take into account the valence of other arbitrary keywords (like lexical affinity), punctuation, and word co-occurrence frequencies. In addition, sophisticated NLP techniques have been developed to address the problems of syntax, negation and irony. Sentiment analysis can be carried out at different levels of text granularity: document (Bosco et al., 2013; Cho et al., 2014; Kontopoulos et al., 2013; Lin et al., 2012; Moraes et al., 2013; Moreo et al., 2012), sentence (Abdul-Mageed et al., 2014; Appel et al., 2016; Desmet & Hoste, 2013; Niu et al., 2016; Patel & Madia, 2016), phrase (Tan et al., 2012), clause, and word (L. Chen et al., 2012; Ghiassi et al., 2013; Quan & Ren, 2014).

Sentiment analysis may be at the sentence or phrase level (which has recently received quite a bit of research attention) or at the document level.

From the perspective of this paper, our work may be seen as document-level sentiment analysis—that is, a document is regarded as an opinion on an entity or aspect of it. This level is associated with the task called document-level sentiment classification, i.e., determining whether a document expresses a positive or negative sentiment.

In (Ravi & Ravi, 2015), the authors presented a survey of over one hundred articles published in the last decade on the tasks, approaches, and applications of sentiment analysis. With a major part of available worldwide data being unstructured (such as text, speech, audio, and video), this poses important research challenges. In recent years numerous research efforts have led to automated SEA, an extension of the NLP area of research. The authors identified seven broad classifications:

1. Subjectivity classification,
2. Sentiment classification,
3. Review usefulness measurement,
4. Lexicon creation,

5. Opinion word and product aspect extraction,
6. Opinion spam detection,
7. Various applications of opinion mining.

The first five dimensions represent tasks to be performed in the broad area of SEA. For the first three dimensions (subjectivity classification, sentiment classification and review usefulness measurement), the authors note that the applied approaches are broadly classified into three categories:

1. Machine learning,
2. Lexicon based,
3. Hybrid approaches.

Since one of our research objectives was to extract sentiment and emotion metadata from documents, the rest of this section focuses on sentiment classification, lexicon creation, and opinion word and product aspect extraction. Sentiment classification is concerned with determining the polarity of a sentence; that is, whether a sentence is expressing positive, negative or neutral sentiment towards the subject. A lexicon is a vocabulary of sentiment words with respective sentiment polarity and strength value while opinion word and product aspect extraction is used to identify opinion on various parts of a product. As per our research objective the rest of the literature review was oriented to document-level sentiment analysis. For our purposes, we assume that a document expresses sentiments on a single content and is written by a single author.

Cho et al. (Cho et al., 2014) proposed a method to improve the positive vs. negative classification performance of product reviews by merging, removing, and switching the entry words of the multiple sentiment dictionaries. They merge and revise the entry words of the multiple sentiment lexicons using labeled product reviews. Specifically, they selectively remove the sentiment words from the existing lexicon to prevent erroneous matching of the sentiment words during lexicon-based sentiment classification. Next, they selectively switch the polarity of the sentiment words to adjust the sentiment values to a specific domain. The remove and switch operations are performed using the target domain's labeled data, i.e. online product reviews, by comparing the positive and negative distribution of the labeled reviews

with a positive and negative distribution of the sentiment words. They achieved 81.8% accuracy for book reviews. However, their contribution is limited to development of a novel method of removing and switching the content of the existing sentiment lexicons.

Moraes et al. (Moraes et al., 2013) compared popular machine learning approaches (SVM and NB) with an ANN-based method for document-level sentiment classification. Naive Bayes (NB) is a probabilistic learning method that assumes terms occur independently while the support vector machine method (SVM) seeks to maximize the distance to the closest training point from either class in order to achieve better generalization/classification performance on test data. The authors reported that, despite the low computational cost of the NB technique, it was not competitive in terms of classification accuracy when compared to SVM. According to the authors, many researchers have reported that SVM is perhaps the most accurate method for text classification. Artificial neural network (ANN) derives features from linear combinations of the input data and then models the output as a nonlinear function of these features. Experimental results showed that, for book datasets, SVM outperformed ANN when the number of terms exceeded 3,000. Although SVM required less training time, it needed more running time than ANN. For 3,000 terms, ANN required 15 sec training time (with negligible running time) while SVM training time was negligible (1.75 sec). In addition, their contribution was limited to performing comparisons between existing approaches. As in (Moraes et al., 2013), Poria S. et al. (Poria et al., 2015) experimented with existing approaches and showed that SVM is a better approach for text-based emotion detection.

2.1.2 Emotion analysis

This section focuses on sentiment and emotion analysis. Emotions include the interpretation, perception and response to feelings related to the experience of any particular situation. Emotions are also associated with mood, temperament, personality, outlook and motivation (Li & Xu, 2014; Munezero et al., 2014; Shivhare & Khethawat, 2012); indeed, the concepts of emotion and sentiment have often been used interchangeably, mostly because both refer to experiences that result from combined biological, cognitive, and social influences. However, sentiments are differentiated from emotions by the duration in which they are experienced.

Emotions are brief episodes of brain, autonomic, and behavioral changes. Sentiments have been found to form and be held over a longer period and to be more stable and dispositional than emotions. Moreover, sentiments are formed and directed toward an object, whereas emotions are not always targeted toward an object.

The emotion-topic model (ETM) (Bao et al., 2012), SWAT model and emotion-term model (ET) (Bao et al., 2012) are the state-of-the-art models. The SWAT model was proposed to explore the connection between the evoked emotions of readers and news headlines by generating a word-emotion mapping dictionary. For each word w in the corpus, it assigns a weight for each emotion e, i.e., P(e|w) is the averaged emotion score observed in each news headline H in which w appears. The emotion-term model is a variant of the NB classifier and was designed to model word-emotion associations. In this model, the probability of word wj conditioned on emotion ek is estimated based on the co-occurrence count between word wj and emotion ek for all documents. The emotion-topic model is combination of the emotion-term model and LDA. In this model, the probability of word wj conditioned on emotion ek is estimated based on the probability of latent topic z conditioned on emotion ek and the probability of word wj conditioned on latent topic z.

A number of techniques exist to detect emotions (Kedar, Bormane, Dhadwal, Alone, & Agarwal, 2015):

1. *Audio based emotion detection*: information from the spectral elements in voice (e.g., speaking rate, pitch, energy of speech, intensity, rhythm regularity, tempo and stress distribution) is used to gather clues about emotions. The features extracted are compared with the training sets in the database using the classifiers;
2. *Blue eyes technology* based on eye moment. In this technique, a picture of the person whose emotions are to be detected is taken and the portion showing his or her eyes is extracted. This extracted image is converted from RGB form to a binary image and compared with ideal eye images depicting various emotions stored in the database. Once the match between the extracted image and one in the database is found, the type of emotion (i.e. happiness, anger, sadness or surprise) is said to be detected;

3. *Facial expression based emotion detection* based on photos of the individual. The images are processed for skin segmentation and analyzed as follows. The image is contrasted, separating the brightest and darkest color in the image area and discriminating the pixels between skin and non-skin. The image is converted into binary form. This processed image is then compared with images forming the training sets in classifiers;
4. *Handwriting based emotion detection* is based on various handwriting indicators or traits of writing (e.g., baseline, slant, pen-pressure, size, zone, strokes, spacing, margins, loops, 'i'-dots, 't'-bar, etc.);
5. *Text based emotion detection* where a computerized NLP approach is used to analyze written text to detect the emotions of the writer. The document is first preprocessed by normalizing the text, then keywords indicating emotional features are extracted. Corresponding emotions are identified through various approaches such as:
    a. Keyword spotting technique,
    b. Lexical affinity method,
    c. Learning based methods,
    d. Hybrid method, or by using an emotion ontology which stores a range of emotion classes, associated keywords and relationships.

Text-based emotion detection approaches focus on 'optimistic', 'depressed' and 'irritated.' The limitations are:

1. Ambiguity of keyword definitions,
2. Inability to recognize sentences without keyword,
3. Difficulty determining emotion indicators.

Lei et al. (Lei et al., 2014) adopted the lexicon-based approach in building the social emotion detection system for online news based on modules of document selection, part-of-speech (POS) tagging, and social emotion lexicon generation. First, they constructed a lexicon in which each word is scored according to multiple emotion labels such as joy, anger, fear, surprise, etc. Next, a lexicon was used to detect social emotions of news headlines. Specifically, given the training set T and its feature set F, an emotion lexicon is generated as a

V×E matrix where the (j, k) item in the matrix is the score (probability) of emotion ek conditioned on feature fj. The authors do not explain how they extracted the features from the document.

Anusha and Sandhya (Anusha & Sandhya, 2015) proposed a system for text-based emotion detection which uses a combination of machine learning and natural language processing techniques to recognize affect in the form of six basic emotions proposed by Ekman. They used the Stanford CoreNLP toolkit to create the dependency tree based on word relationships. Next, phrase selection is done using the rules on dependency relationships that gives priority to the semantic information for the classification of a sentence's emotion. Based on the phrase selection, they used the Porter stemming algorithm for stemming, and stopwords removal and tf-idf to build the feature vectors. The authors do not propose a new approach but implement existing algorithms.

Cambria et al. (Cambria et al., 2015) explored how the high generalization performance, low computational complexity, and fast learning speed of extreme learning machines can be exploited to perform analogical reasoning in a vector space model of affective common-sense knowledge. After performing TSVD on AffectNet, they used the Frobenius norm to derive a new matrix. For the emotion categorization model, they used the Duchenne smile and the Klaus Scherer model. As in (Anusha & Sandhya, 2015), the authors do not propose a new approach but implement existing algorithms.

2.1.3 Conclusion

Following is our conclusions on related work in sentiment and emotion analysis:

1. Traditional sentiment analysis methods mainly use terms and their frequency, part of speech, rule of opinions and sentiment shifters. Semantic information is ignored in term selection, and it is difficult to find complete rules;
2. Most of the recent contributions are limited to sentiment analysis elaborated in terms of positive or negative opinion and do not include analysis of emotion;

3. Existing approaches do not take into account the human contribution to improve accuracy;
4. Existing approaches do not combine sentiment and emotion analysis;
5. Lexicon and ontology based approaches provide good accuracy for text-based sentiment and emotion analysis when applying SVM techniques. In our work, it is more important to identify the sentiment and emotion of a book taking into account all the books of the collection. For example, assume that book A has 90% fear and 80% sadness while the emotion which has the best weight of book B is 40% fear; can it be said that fear is the emotion of book B as in book A? ;
6. Existing approaches do not take into account document collections. In terms of granularity, most of the existing approaches are sentence-based;
7. These approaches do not take into account the context around the sentence and in this way, it is possible to lose the real emotion.

As a general conclusion to the literature review on topic detection, sentiment and emotion analysis, 95% of the work focused on features of the documents (e.g., sentence length, capitalized words, document title, term frequency, and sentences position) to perform text mining and generally make use of existing algorithms or approaches (e.g., LDA, tf-idf, VSM, SVD, LSA, TextRank, PageRank, LexRank, FCA, LTM, SVM, NB and ANN) based on their associated features to documents.

Table A 2.1 compares the most known text mining algorithms (e.g., AlchemyAPI, DBpedia, Wikimeta, open calais, Bitext, AIDA, TextRazor) with our proposed algorithms in SMESE V3 by keyword extraction, classification, sentiment analysis, emotion analysis and concept extraction.

Table A 2.1 Summary of attribute comparison of existing and proposed SMESE V3 algorithms

Existing algorithms	Keyword extraction	Classification	Sentiment analysis	Emotion analysis	Concept extraction
AlchemyAPI (http://www.alchemyapi.com/)	x		x	x	x
DBpedia Spotlight (https://github.com/dbpedia-spotlight)					x
Wikimeta (https://www.w3.org/2001/sw/wiki/Wikimeta)					x
Yahoo! Content Analysis API (out of date) (https://developer.yahoo.com/contentanalysis/)		x			x
Open Calais (http://www.opencalais.com/)	x	x			x
Tone Analyzer (https://tone-analyzer-demo.mybluemix.net/)			x	x	
Zemanta (http://www.zemanta.com/)					x
Receptiviti (http://www.receptiviti.ai/)			x	x	
Apache Stanbol (https://stanbol.apache.org/)					x
Bitext (https://www.bitext.com/)			x		x
Mood patrol (https://market.mashape.com/soulhackerslabs/moodpatrol-emotion-detection-from-text)				x	
Aylien (http://aylien.com/)	x	x	x		
AIDA (http://senseable.mit.edu/aida/)					x
Wikifier (http://wikifier.org/)					x
TextRazor (https://www.textrazor.com/)					x
Synesketch (http://krcadinac.com/synesketch/)				x	
Toneapi (http://toneapi.com/)			x	x	
SMESE V3	x	x	x	x	x

3. Rule-based semantic metadata internal enrichment

This section presents an overview and details of the proposed rule-based semantic metadata internal enrichment (RSMIE), including two algorithms (BM-SATD and BM-SSEA) used to process semantic metadata internal enrichment.

RSMIEE is part of the SMESE V3 platform architecture as shown in Figure A. 2.1. The main goal of this paper is to enhance the SMESE platform through text analysis approaches for topics, sentiment and emotion and semantic relationships detection. SMESE V3 allows one to create a semantic master catalogue with enriched metadata (e.g., topics, sentiments and emotions) that enables the search and discovery interest-based processes. To perform this task, the following tools are needed:

1. Topics are a controlled set of terms designed to describe the subject of a document. While topics do not necessarily include relationships between terms, we include relationships as triplets (Entity – Relationship – Entity); for example, Entity "Ronald" - relationship:" likes " - Entity "Le petit prince";
2. A multilingual thesauri and ontology to provide hierarchical relationships as well as semantic relationships between topics;
3. An ontology to provide a representation of knowledge with rich semantic relationships between topics. By breaking content into pieces of data, and curating semantic relationships to external contents, metadata enrichments are created dynamically.

In Figure A. 2.1, the improvements to the SMESE platform from this work and its implementation are presented in blue.

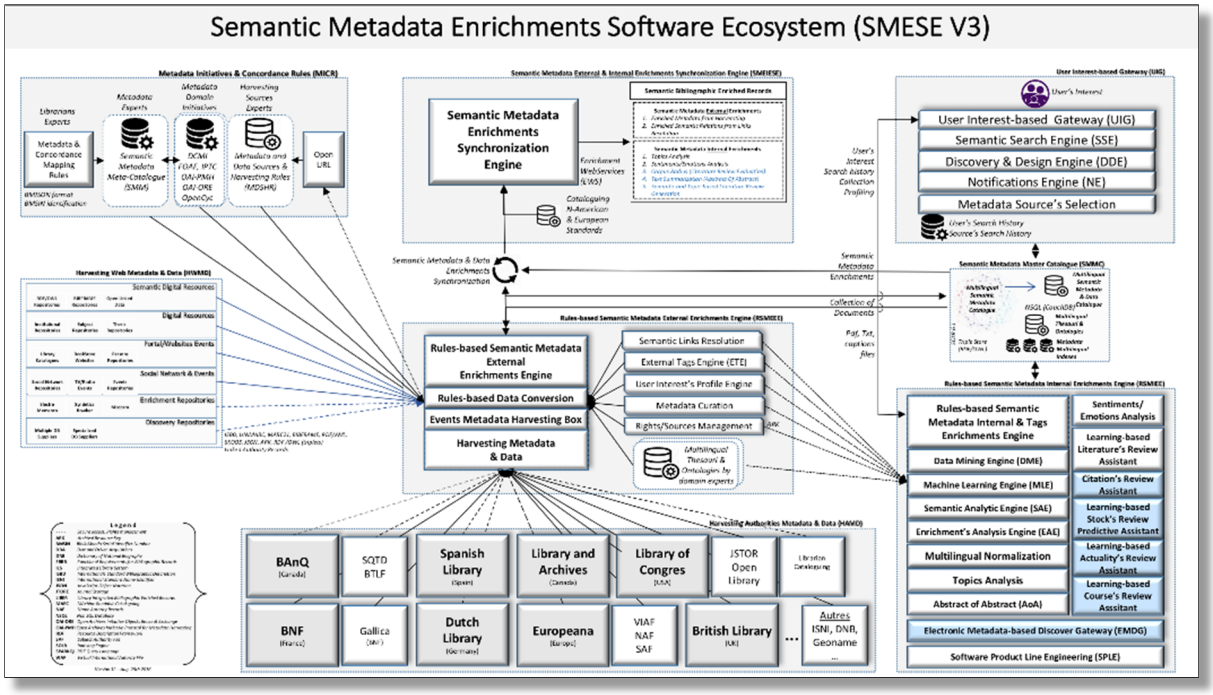


Figure A 2.1 SMESE V3 –Semantic Metadata Enrichment Software Ecosystem

3.1 RSMIEE overview

RSMIEE has been designed to find short descriptions, in terms of topics, sentiments and emotions of the members of a collection to enable efficient processing of large collections while preserving the semantic and statistical relationships that are useful for tasks such as: topic detection, classification, novelty detection, summarization, and similarity and relevance judgments. Figure A 2.2 shows an overview of the architecture of RSMIEE that consists of:

1. User interest-based gateway,
2. Metadata initiatives & concordance rules,
3. Harvesting web metadata & data,
4. User profiling system,
5. Rule-based semantic metadata internal enrichment.

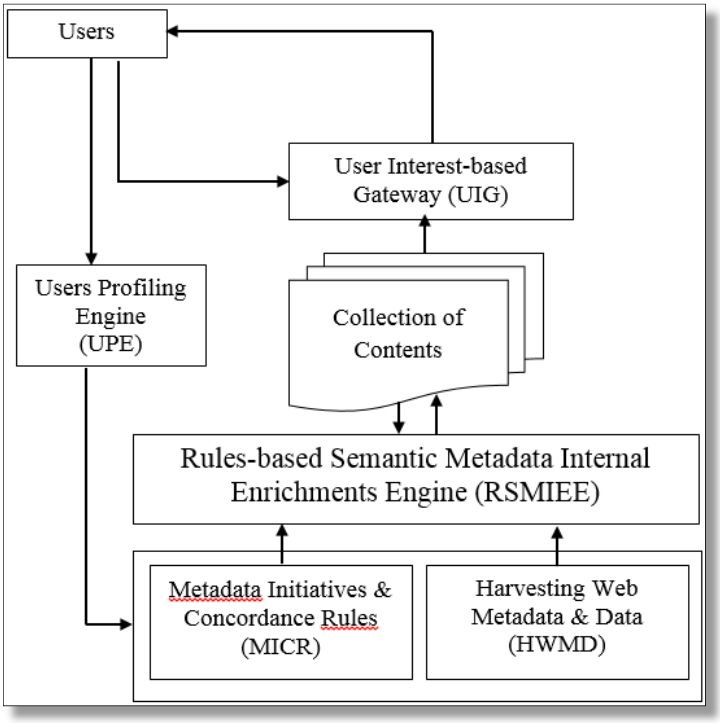


Figure A 2.2 Overview of the RSMIEE architecture

The user interest-based gateway (UIG) is designed to push notifications to users based on the emotions and interests found using the user-profiling system (UPS). UIG is also a discovery tool that allows users to search and discover contents based on their interests and emotions.

The user-profiling system (UPS) applies machine learning algorithms to user feedback in terms of appreciation, rating, comment and historical research in order to provide user profiles. When the contextual information of users is available, it is used to increase the accuracy of the profiling process.

RSMIEE performs automated metadata internal enrichment based on the set of metadata initiatives & concordance rules (MICR), the process for harvesting web metadata & data (HWMD), the user profile and a thesaurus. RSMIEE implements BM-SATD for topic automated detection from documents and BM-SSEA is implemented for sentiment and emotion detection of documents.

BM-SATD and BM-SSEA tasks may be redefined as document classification issues as they contain methods for the classification of natural language text. That is, methods that will predict the query's category, given a set of training documents with known categories and a new document, which is usually called the query.

The following sub-sections present the terminology and assumptions, the necessary pre-processing and details of the two algorithms implemented in RSMIEE.

3.2 Terminology and assumptions

In this section the following terms are defined:

1. A word or term is the basic unit of discrete data, defined to be an item from a vocabulary indexed by {1, …,V}. Terms are presented using unit-basis vectors that have a single component equal to one and all other components equal to zero. Thus, using superscripts to denote components, the ith term in the vocabulary is represented by an I-vector w such that $w_i = 1$ and $w_j = 0$ for $i \neq j$. For example, let V= {book, image, video, cat, dog} be the vocabulary. The video term is represented by the vector (0, 0, 1, 0, 0);
2. A line is a sequence of N terms denoted by l. These terms are extracted from a real sentence; a sentence is a group of words, usually containing a verb, that expresses a thought in the form of a statement, question, instruction, or exclamation and when written begins with a capital letter;
3. A document is a sequence of N lines denoted by D = (w_1,w_2; …,w_N), where wi is the ith term in the sequence coming from the lines. D is represented by its lines as D = (l_1, …l_i,…,l_K);
4. A corpus is a collection of M documents denoted by C = {D_1, D_2, …, D_M};
5. An emotion word is a word with strong emotional tendency. An emotion word is a probabilistic distribution of emotions and represents a semantically coherent emotion analysis. For example, the word "excitement", presenting a positive and pleased feeling, is assigned a high probability to emotion "joy".

To implement the BM-SATD and BM-SSEA algorithms, an initial set of conditions must be established:

1. A list of topics $T = \{t_1, \dots, t_i, \dots, t_n\}$ is readily available;
2. Each existing document Dj is already annotated by topic. The annotated topics of document Dj are denoted as $TDj = \{t_p \dots, t_i, \dots, t_q\}$ where tp, ti, and $tq \in T$;
3. The corpus of documents is already classified by topics. Cti={…,Dj,…} denotes the corpus of documents that have been annotated with topic ti. Note that the document Dj may be located in several corpuses;
4. A list of emotions $E = \{e_1, \dots, e_i, \dots, e_E\}$ is readily available with the common instances of e being joy, anger, fear, surprise, touching, empathy, boredom, sadness, warmth;
5. A set of ratings over E emotion labels denoted by $R_{Dj} = \{rd_{,e1} \dots, r_{d,ei}, \dots, r_{d,eE}\}$. The value of rd,ei is the number of users who have voted ith emotion label ei for document d. In other words, rd,ei is the number of users who claimed that emotion ei is found in document d;
6. The corpus of documents are already classified by sentiment and emotion based on the user rating. Cei = {…,Dj,…} denotes the corpus of documents rated with emotion ei. Note that the document Dj may be located in several knowledge corpus;
7. A list of sentiments $S = \{s_1, \dots, s_i, \dots, s_S\}$ is readily available;
8. A thesaurus is available and has a tree hierarchical structure. A thesaurus contains a list of words with synonyms and related concepts. This approach uses synonyms or glosses of lexical resources in order to determine the emotion or polarity of words, sentences and documents.

3.3 Document pre-processing

Before document analysis, RSMIEE performs a pre-processing. The objective of the pre-processing is to filter noise and adjust the data format to be suitable for the analysis phases. It consists of stemming, phase extraction, part-of-speech filtering and removal of stop-words. The corpus of documents crawled from specific databases or the internet consists of many

documents. The documents are pre-processed into a basket dataset C, called document collection. C consists of lines representing the sentences of the documents. Each line consists of terms, i.e. words or phrases. An example of C follows:

C=

……

……

D_j= (line 1) : term 1, term 2, term 6, term 9.

(line 2) : term 10, term 6, term 2, term 3.

……

(line i) : term 3, term 5, term 2, term 3, term 9, term 1.

……

(line N_j) : term 2, term 15, term 9, term 3, term 4.

……

……

More specifically, to obtain Dj, the following preprocessing steps are performed:

1. Language detection;
2. Segmentation: a process of dividing a given document into sentences;
3. Stop word: a process to remove the stop words from the text. Stop words are frequently occurring words such as 'a' an', the' that provide less meaning and generate noise. Stopwords are predefined and stored in an array;
4. Tokenization: separates the input text into separate tokens;
5. Punctuation marks: identifies and treats the spaces and word terminators as the word breaking characters;
6. Word stemming: converts each word into its root form by removing its prefix and suffix for comparison with other words.

More specifically, a standard preprocessing such as tokenization, lowercasing and stemming of all the terms using the Porter stemmer (Porter, 1980). Therefore, we also parse the texts using the Stanford parser (de Marneffe M-C, MacCartney B, & Manning CD, 2006) that is a

lexicalized probabilistic parser which provides various information such as the syntactic structure of text segments, dependencies and POS tags.

‘Word’ and ‘term’ are used interchangeably in the rest of this paper.

3.4 Scalable annotation-based topic detection: BM-SATD

The aim of BM-SATD is to build a classifier that can learn from already annotated contents (e.g., documents and books) and infer the topics of new books. Traditional approaches are typically based on various topic models, such as latent Dirichlet allocation (LDA) where authors cluster terms into a topic by mining semantic relations between terms. However, co-occurrence relations across the document are commonly neglected, which leads to detection of incomplete information. Furthermore, the inability to discover latent co-occurrence relations via the context or other bridge terms prevents important but rare topics from being detected. BM-SATD combines semantic relations between terms and co-occurrence relations across the document making use of document annotation. In addition, BM-SATD includes:

1. A probabilistic topic detection approach that is an extension of LDA, called BM semantic topic model (BM-SemTopic);
2. A clustering approach that is an extension of KeyGraph, called BM semantic graph (BM-SemGraph).

BM-SATD is a hybrid relation analysis and machine learning approach that integrates semantic relations, semantic annotations and co-occurrence relations for topic detection. More specifically, BM-SATD fuses multiple relations into a term graph and detects topics from the graph using a graph analytical method. It can detect topics not only more effectively by combing mutually complementary relations, but also mine important rare topics by leveraging latent co-occurrence relations.

BM-SATD is composed of five phases:

1. Relevant and less similar documents selection process phase,
2. Not annotated documents semantic term graph generation process phase,
3. Topics detection process phase,

4. Training process phase,
5. Topics refining process phase.

The following sub-sections present the details of the five phases of the BM-SATD model.

3.4.1 Relevant and less similar documents selection process phase

For a given topic, a filtering process is performed to avoid using a large corpus of documents that are similar or not relevant. It is not necessary to compare a new document of a collection with two other documents of the collection that are similar in order to know whether this new document is similar to each of the other documents. This strategy merely increases computation time. For this reason, only relevant and less similar documents within a corpus are identified. Here, only documents that are already annotated by topic are considered.

An overview of the architecture of the relevant and less similar document selection phase is presented in Figure A 2.3. This phase involves three algorithms:

1. Algo 1 identifies the relevant documents for a given topic;
2. Algo 2 detects less similar documents in the relevant set of documents;
3. Algo 3 ascertains whether the new annotated document with a topic is relevant and less similar to a sub set of relevant and less similar documents of this topic.

First, the most relevant documents of each topic ti are selected. For each document of a topic ti, Algo 1 checks whether its most important terms are the same as the most important terms of the topic t_i. To identify the most important terms of a given document D_j, the tf-idf of each term Wi in the corpus C_{ti} is computed using equation (A 2.1):

$$f(W_i, D_j, C_{ti}) = TF - IDF(W_i, D_j, C_{ti}) \qquad \text{(A 2.1)}$$
$$= TF(W_i, D_j) * \log(\frac{|C_{ti}| = M_i}{IDF(W_i, C_{ti})})$$

where $TF(W_i, D_j)$, $IDF(W_i, C_{ti})$ and M_i denote the number of occurrences of W_i in document D_j, the number of documents in the corpus C_{ti} where W_i appears, and the number of documents in the corpus C_{ti}, respectively.

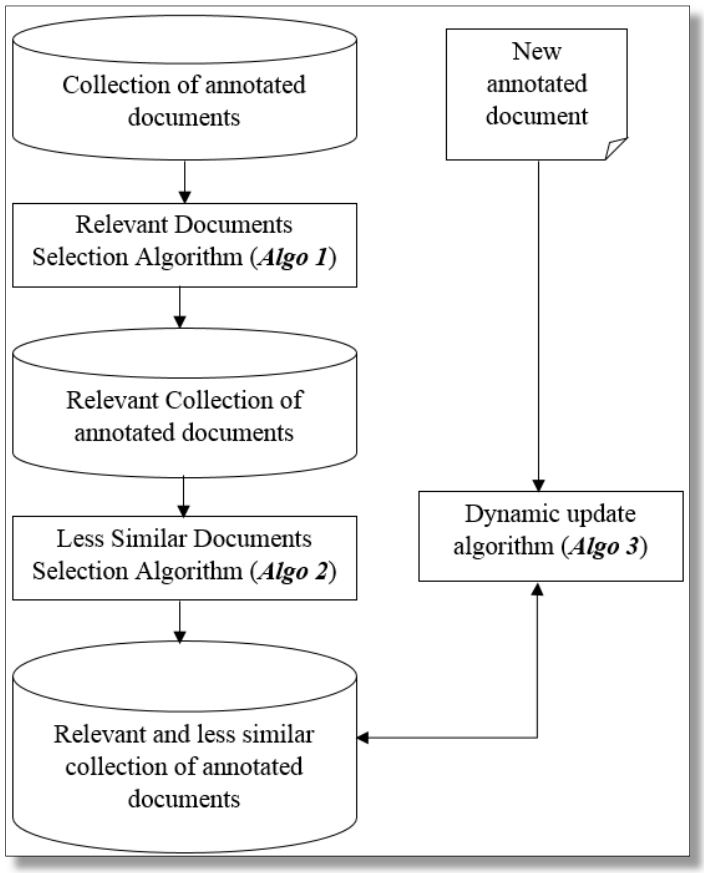


Figure A 2.3 Relevant and less similar document selection process phase – Architecture overview

Equation (A 2.1) allows BM-SATD to find, for each document D_j, the vector V_{Dj}= { $(W_a, f(W_a,D_j,C_{ti}))$, ..., $(W_i, f(W_i,D_j,C_{ti}))$,..., $(W_{|Dj|}, f(W_{|Dj|},D_j,C_{ti}))$} where in the couple $(W_i, f(W_i,D_j,C_{ti}))$, W_i denotes a term and $f(W_i,D_j,C_{ti}))$ its tf-idf in the whole corpus C_{ti}..

To identify the most important terms of a given topic t_i, the tf-itf of each term W_k that appears at least one time in at least one document of corpus C_{ti} is computed with formula (A 2.2):

$$g(W_k, t_i) = TF - ITF(W_k, t_i) = TF(W_k, t_i) * \log(\frac{|T| = n}{ITF(W_k)}) \quad \text{(A 2.2)}$$

where $TF(W_k, t_i)$, $ITF(W_k)$ and |T| denote the number of occurrences of W_k in all the documents of corpus C_{ti}, the number of topics where W_k appears, and the number of topic, respectively.

Equation (A 2.2) allows BM-SATD to find, for each topic t_i, the vector V_{ti}= { $(W_1, g(W_1,t_i))$, …, $(W_k, g(W_k,t_i))$,…, $(W_{Ni}, g(W_{Ni},t_i))$} where in the couple $(W_k, g(W_k,t_i))$, W_k denotes a term and $g(W_k,t_i)$ its tf-itf in the whole corpus T.

Let N_i be the number of terms of the vocabulary of C_{ti} and N_{Dj} = |Dj| be the number of terms of the vocabulary of D_j. In this context, N_i is larger than N_{Dj}. To determine the number of terms to consider the document relevant, BM-SATD computes the standard deviation σ and the average *avg* of the number of distinct terms in the documents for the topics. BM-SATD uses the standard deviation because it gives a good indication of the dispersion of data from the average. The standard deviation σ_{ti} of topic t_i is given by equation (A 2.3):

$$\sigma_{ti} = \sqrt{\frac{\sum_{j=1}^{|C_{ti}|=M_i}(|D_j| - avg_{ti})^2}{|C_{ti}| = M_i}} \quad \text{(A 2.3)}$$

where the average number of terms avg_{ti} of topic t_i is computed using equation (A 2.4).

$$avg_{ti} = \frac{\sum_{j=1}^{|C_{ti}|=M_i}|D_j|}{|C_{ti}| = M_i} \quad \text{(A 2.4)}$$

Next, to compute the number of distinct terms to consider, BM-SATD uses equation (A 2.5).

$$E_{ti} = avg_{ti} - \sigma_{ti} \quad \text{(A 2.5)}$$

The score for each document D_j in the topic t_i is computed next:

1. BM-SATD sorts, for each document D_j of corpus C_{ti}, the vector V_{Dj} by $f(W_i,D_j,C_{ti})$ in descending order;

2. BM-SATD computes the BMscore of D_j using equation (A 2.6):

$$BMscore\left(D_j\right) = \sum\nolimits_{|E_i|} g(W_i, t_i) \tag{A 2.6}$$

where $\sum_{|Ei|}$ are the first $|E_i|$ terms W_i of D_j with the highest value of $f(W_i,D_j,C_{ti})$ in the whole corpus C_{ti} .

In order terms, BMscore is the summation of the tf-itf in the whole corpus C of the first $|E_i|$ terms W_i of D_j with the highest tf-idf in the whole corpus C_{ti}.

Finally, based on the BMscore of each document D_j of corpus C_{ti}, BM-SATD selects the most relevant documents of corpus C_{ti}. BM-SATD obtains the sub-corpus C'_{ti} of the most relevant documents using equation (A 2.7):

$$C_{ti} = \left[C'_{ti} = \bigcup_{\alpha}\{D_k\}\right] \cup \left[\bigcup_{M_i-\alpha}\{D_j\}\right] \tag{A 2.7}$$

where BMscore (Dk) > BMscore (Dj).

Note that $\boldsymbol{\alpha}$ is a threshold determined by empirical experimentation based on the particular document collection. $C'_{ti} = \{D_{k_1}, \dots, D_{k_i}, \dots, D_{k_\alpha}\}$ is obtained where $M_i > M'_i = \alpha$. Algorithm 1 of appendix A explains, in detail, the selection process of relevant documents for a given topic.

The less similar documents of sub-corpus C'_{ti} for the topic t_i are then selected. BM-SATD defines a similarity threshold $\boldsymbol{\beta}$ by empirical experimentation based on the particular document collection where C''_{ti} is the sub-corpus of C'_{ti} that contains the less similar documents.

1. BM-SATD sorts the documents of C'_{ti} according to their BMscore. BM-SATD first puts the document with the largest BMscore in C''_{ti}; then, based on the order of largest BMscore, BM-SATD compares the semantic similarity of each element of C''_{ti} with the rest of element of $C'_{ti}C'_{ti}$. If no document of C''_{ti} is semantically similar to a given

document of C'_{ti}, this given document is added to C''_{ti}. When the semantic similarity between two documents is less than or equal to $\boldsymbol{\beta}$, BM-SATD assumes they are not similar. Algorithm 2 of appendix A gives more detail about the selection process of less similar documents for a given corpus that allows one to obtain the sub-corpus $C''_{ti} = \left\{D_{k_1}, \dots, D_{k_l}, \dots, D_{k_\gamma}\right\}$ where $\alpha \geq \gamma$;

2. Finally, when a new document annotated with topic t_i, is added to the corpus C_{ti}, BM-SATD computes its BMscore in order to ascertain whether this new document must be added to C''_{ti} or not.

For example, let IDF^s_{ti} be the idf vector of the vocabulary of corpus C_{ti} at state *s* and ITF^s be the itf vector of the vocabulary of corpus C at state *s*. The state is the situation of the collection before adding the new document: $IDF^s_{ti} = (IDF(W_1, C_{ti}), \dots, IDF(W_k, C_{ti}), \dots, IDF(W_{Ni}, C_{ti}))$ and $ITF^s = (ITF(W_1), \dots, ITF(W_k), \dots, ITF(W_{Ni}))$. Let TF^s_{ti} be the tf vector of the vocabulary of corpus C_{ti} at the state *s*: $TF^s_{ti} = (TF(W_1, t_i), \dots, TF(W_k, t_i), \dots, TF(W_{Ni}, t_i))$.

Based on vector IDF^s_{ti}, BM-SATD computes the TF-IDF of each term *W* of *d* of each term w of *d* using equation (A 2.8):

$$f(W, d, C_{ti}) = TF - IDF(W, d, C_{ti}) \tag{A 2.8}$$
$$= TF(W, d) * \log\left(\frac{|C_{ti}|}{IDF(W, C_{ti}) + 1}\right)$$

Next, BM-SATD ranks the vocabulary of d according to their $f(W, d, C_{ti})$ and selects the E_{ti} terms *W* of d with highest $f(W, d, C_{ti})$. Based on the vectors ITF^s_{ti} and TF^s_{ti}, BM-SATD computes the TF-ITF of each selected term *W* of *d* using equation (A 2.9):

$$g(W, t_i) = TF - ITF(W, t_i) \tag{A 2.9}$$
$$= [TF(W, t_i) + TF(W, d)] * log\left(\frac{|T|}{ITF(W_k)}\right)$$

BM-SATD obtains the BMscore(d) of new document *d* by summation of the g(W,t_i) term. If BMscore(d) is greater than the smallest BMscore of C'_{ti} document, BM-SATD uses Algorithm 2 to make a semantic similarity computation and then performs an update of C''_{ti} if necessary. Algorithm 3 of appendix A presents the C''_{ti} update process of a given corpus t_i.

3.4.2 Not annotated documents semantic term graph generation process phase

The semantic term graph allows one to convert a set of lines of terms into a graph by extracting semantic and co-occurrence relations between terms. The semantic term graph is a basis for detecting topics automatically.

To generate the semantic term graph BM-SemGraph:

1. First the co-occurrence clusters are generated and then optimized;
2. After cluster optimization, the keys terms and key links between the clusters are extracted;
3. Finally, the semantic topic is generated and semantic term graph extracted.

The BM-SemGraph has one node for each term in the vocabulary of the document. Edges in a BM-SemGraph represent the co-occurrence of the corresponding keywords and are weighted by the count of the co-occurrences.

Note that, in contrast to existing graph-based approaches, the co-occurrence between A and B is different from the co-occurrence between B and A. This difference allows one to retain the semantic sense of co-occurrence terms. Figure A 2.4 presents an overview of the architecture of the semantic term graph generation process phase. Two sub processes (the term graph process and BM-SemTopic process) generate the semantic graph in order to enrich the term graph with semantic information; indeed, the terms graph and semantic graph are merged to provide Semantic term graph, called BM-SemGraph.

The term graph process consists of three steps:

1. Co-occurrence clusters generation,
2. Clusters optimization,

3. Key terms extraction.

The BM-SemTopic process consists of two steps:

1. Semantic topic generation,
2. Semantic graph extraction.

Step 1: Co-occurrence clusters generation

For the co-occurrence graph, the assumption is that terms that have a close relation to each other may be linked by the co-occurrence link. The relation between two terms W_i and W_j is measured by their conditional probability. Let D be a document and $V_D = (w_1, w_2; \ldots, w_N)$ be the terms of D and L_D be the number of lines of D.

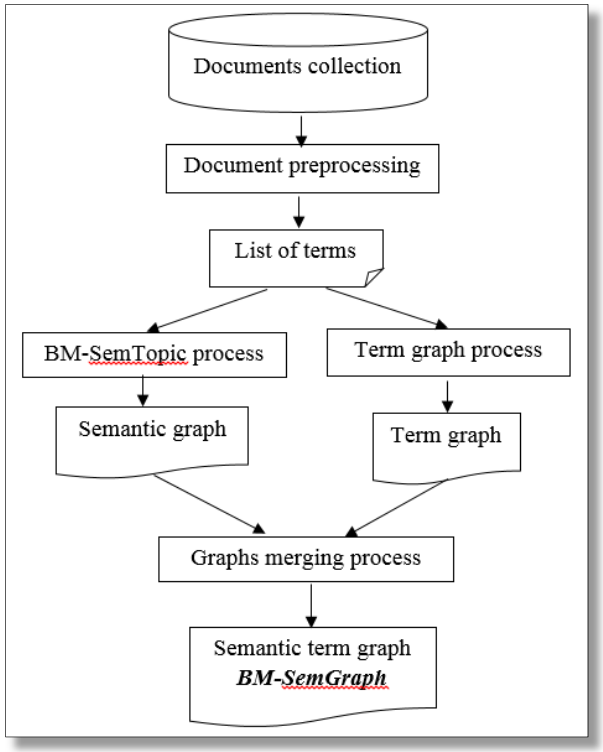


Figure A 2.4 New document semantic term graph process phase - Architecture overview

The conditional probability $p(\overrightarrow{W_i, W_j}^{\varepsilon})$ of $\overrightarrow{W_i, W_j}^{\varepsilon}$ is computed using equation (A 2.10) where:

1. ε denotes the minimum distance between W_i and W_j;

2. The distance between two terms is the number of terms that appear between them for a given line;
3. ε is a parameter determined by experimentation.

$$p\left(\overrightarrow{W_i, W_j}^{\varepsilon}\right) = \sum_{l=1}^{L_D} \frac{N^{line\ l}(\overrightarrow{W_i, W_j}^{\varepsilon})}{\left\lceil \frac{N(line\ l)}{\varepsilon} \right\rceil} \tag{A 2.10}$$

where $N^{line\ l}\left(\overrightarrow{W_i, W_j}^{\varepsilon}\right)$ denotes the number of times that W_i and W_j co-occur with a minimum distance ε and where W_i appears before W_j, and *N(line l)* denotes the number of terms of the line l.

To formally define a relation between two terms W_i and W_j, their frequent co-occurrence measured by the conditional probability $p(\overrightarrow{W_i, W_j}^{\varepsilon})$, needs to exceed the co-occurrence threshold. The co-occurrence threshold is also determined by experimentation. Note that frequent co-occurrence is oriented. This allows one to retain the semantic orientation of the links between terms.

Next, the oriented links are transformed into simple links without losing the semantic context. To perform this transformation, three rules are applied - see Figure A 2.5.

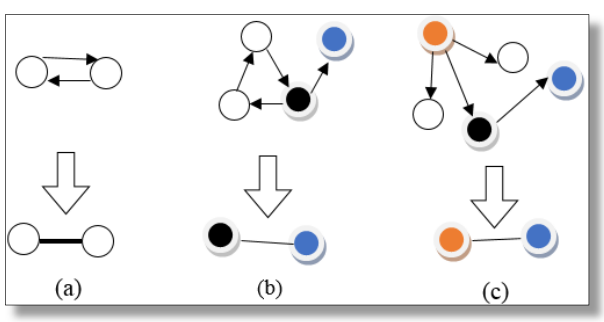


Figure A 2.5 Link transformation rules

In Figure A 2.5a, two nodes with two oriented links are transformed into one simple link. In this case, this type of link cannot be pruned and its weight is given by equation (A 2.11):

$$w(W_i, W_j) = p\left(\overrightarrow{W_i, W_j}^{\varepsilon}\right) + p\left(\overrightarrow{W_j, W_i}^{\varepsilon}\right) \quad \text{(A 2.11)}$$

In Figure A 2.5b, where several nodes are linked by oriented links and there is an oriented path to join each of them, only the nodes with a link to other nodes not in the oriented path are retained. This is the situation of the black node and blue node. The black node becomes the representative of the other nodes.

In Figure A 2.5c, where one node A is linked to several nodes and the links are oriented from A towards the other nodes, node A becomes the representative of the other nodes and the other nodes are removed. This is the case for the red node where the link between the black node and blue node is removed and a new link is added between the red node and the blue node.

Let *G* be a set of nodes where W_i is the representative node. Let *G'* be the sub set of *G* which are linked to a node W_j not in G. Figure A 2.6 illustrates the representation of G and G'.

The weight of the link between W_i and W_j is given by equation (A 2.12):

$$w(W_i, W_j) = \sum_{W_k \in G'} p\left(\overrightarrow{W_k, W_j}^{\varepsilon}\right) + p\left(\overrightarrow{W_j, W_k}^{\varepsilon}\right) \quad \text{(A 2.12)}$$

Equation (A 2.12) is applied in the case of Figure A 2.4b and Figure A 2.4c to compute the weight of the link between a representative node and another node. Finally, the rest of the oriented links are transformed into simple links and their weights computed using equation (A 2.11).

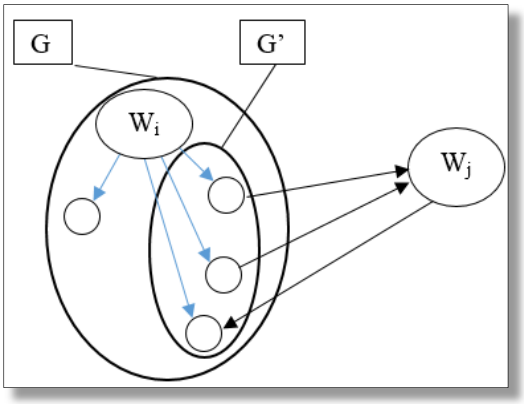


Figure A 2.6 Representation of the computation of weight after removing some nodes

Step 2: Cluster optimization

To enhance quality, clusters should be pruned, such as by removing weak links or partitioning sparse cluster into cohesive sub-clusters. Clusters are pruned according to their connectedness. The link *e* is pruned when no path connects the two ends of *e* after it is pruned. As shown in Figure A 2.7, the link between the black node and the green node should be pruned.

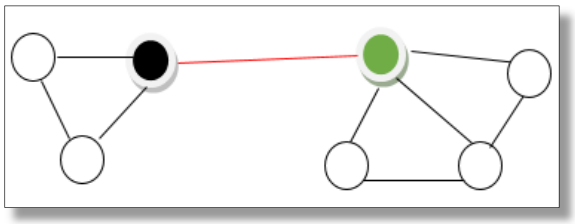


Figure A 2.7 Clusters optimization

Secondly, cliques are identified. In graph theory, a clique is a set of nodes which are adjacent pairs as shown in Figure A 2.8.

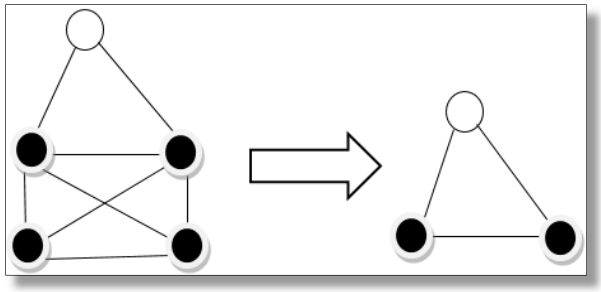


Figure A 2.8 Clique reduction

Let C be the clique and W_i and W_j be the nodes of C that are linked to another node. The weight between W_i and W_j is given by equation (A 2.13):

$$w(W_i, W_j) = \underset{\substack{W_k \in C \\ W_s \in C}}{\mathrm{MAX}}[w(W_k, W_s)] \tag{A 2.13}$$

Step 3: Key term extraction

To extract key terms, the relation between a term and a cluster is measured. It is assumed that the weight of a term in a given cluster may be used to determine the importance of this term for the cluster. Let R be the set of nodes of the cluster C where the node W_i is inside. The weight of W_i in the cluster C is given by equation (A 2.14):

$$f(W_i) = \sum_{W_j \in R} w(W_i, W_j) \tag{A 2.14}$$

To identify a term as a key term, a sort of terms is performed based on their weights regardless of the clusters that they are in. Next, the NumKeyTerm terms that have the largest weights are selected as Key Terms. NumKeyTerm is a parameter.

Step 4: Semantic topic generation

Semantic topic generation combines a correlated topic model (CTM) (David M. Blei & Lafferty, 2005) and a domain knowledge model (DKM) (Andrzejewski, Zhu, & Craven, 2009), called BM semantic topic model (BM-SemTopic), to build the real semantic topic model. In LDA, a topic is a probability distribution over a vocabulary. It describes the relative frequency

each word is used in a topic. Each document is regarded as a mixture of multiple topics and is characterized by a probability distribution over the topics.

A limitation of LDA is its inability to model topic correlation. This stems from the use of the Dirichlet distribution to model the variability among topic proportions. In addition, standard LDA does not consider domain knowledge in topic modeling.

To overcome these limitations, BM-SemTopic combines two models:

1. A correlated topic model (CTM) (David M. Blei & Lafferty, 2005) that makes use of a logistic normal distribution;
2. A domain knowledge model (DKM) (Andrzejewski et al., 2009) that makes use of the Dirichlet distribution.

BM-SemTopic uses a weighted sum of CTM and DKM to compute the probability distribution of term W_i on the topic z. The sum is defined by equation (A 2.15):

$$h(W_i|\mathrm{z}) = \omega CTM(W_i|\mathrm{z}) + (1-\omega)\, DKM(W_i|\mathrm{z}) \qquad \text{(A 2.15)}$$

where ω is used to give more influence to one model based on the term distribution of topics.

When the majority of terms are located in a few topics, this means the domain knowledge is important and ω must be small. BM-SemTopi*c* develops the CTM where the topic proportions exhibit a correlation with the logistic normal distribution and incorporates the DKM. A key advantage of BM-SemTopic is that it explicitly models the dependence and independence structure among topics and words, which is conducive to the discovery of meaningful topics and topic relations.

CTM is based on a logistic normal distribution. The logistic normal is a distribution on the simplex that allows for a general pattern of variability between the components by transforming a multivariate normal random variable. This process is identical to the generative process of LDA except that the topic proportions are drawn from a logistic normal distribution rather than a Dirichlet distribution. The strong independence assumption imposed by the Dirichlet in LDA is not realistic when analyzing document collections where one may find

strong correlations between topics. To model such correlations, the covariance matrix of the logistic normal distribution in the BM-SemTopic correlated topic model is introduced.

DKM is an approach to incorporation of such domain knowledge into LDA. To express knowledge in an ontology, BM-SemTopic uses two primitives on word pairs: Links and Not-Links. BM-SemTopic replaces the Dirichlet prior by the Dirichlet Forest prior in the LDA model. Then, BM-SemTopic sorts the terms for every topic in descending order according to the probability distribution of the topic terms. Next it picks up the high-probability terms as the feature terms. For each topic, the terms with probabilities higher than half of the maximum probability distribution are picked up (experiment indicates it is non-sensitive on this parameter).

Step 5: Semantic term graph extraction

To enrich the term graph, the semantic topic needs to be converted into a semantic graph that consists of semantic relations between the semantic terms. To discover these relations, the semantic aspect is included making use of WordNet::Similarity (Pedersen, Patwardhan, & Michelizzi, 2004). Based on the structure and content of the lexical database WordNet, WordNet::Similarity implements six measures of similarity and three measures of relatedness. Measures of similarity use information found in a hierarchy of concepts (or synsets) that quantify how much concept A is like (or is similar to) concept B.

First, each generated feature term at step 4 is the candidate for a semantic term where it is assumed the other terms represent the vocabulary associated with the semantic topic. In Figure A 2.9a, the blue node denotes the feature terms of each semantic topic.

Next, duplicate terms from the candidates are removed. If there is more than one topic that has the same term W_j in the semantic term candidate, only the topic z with the highest term probability distribution $h(W_j|z)$ is retained W_j as the semantic term candidate. It follows then that following this step the semantic term candidates of different topics are exclusive to each other. Figure A 2.9b shows the remaining candidates by semantic topic.

To remove similar terms, the measure path (one measure of similarity of WordNet::Similarity (Pedersen et al., 2004)) is used to evaluate similarity between two terms. The measure path of WordNet::Similarity is a baseline that is equal to the inverse of the shortest path between two concepts. When the semantic term candidates of different topics are identified, the semantic value of each topic's candidates is computed. The semantic value of each term W_i, is given by equation (A 2.16):

$$SEM(W_i|\mathrm{z}) = TP - ITP(W_k|\mathrm{z}) = h(W_i|\mathrm{z}) * \log\left(\frac{|Z|}{\sum_{t \in Z} h(W_i|\mathrm{t})}\right) \qquad \text{(A 2.16)}$$

where Z denotes the set of semantic topics. TP-ITP is inspired by the tf-idf formula, where TP is term probability and ITP inverse topic probability.

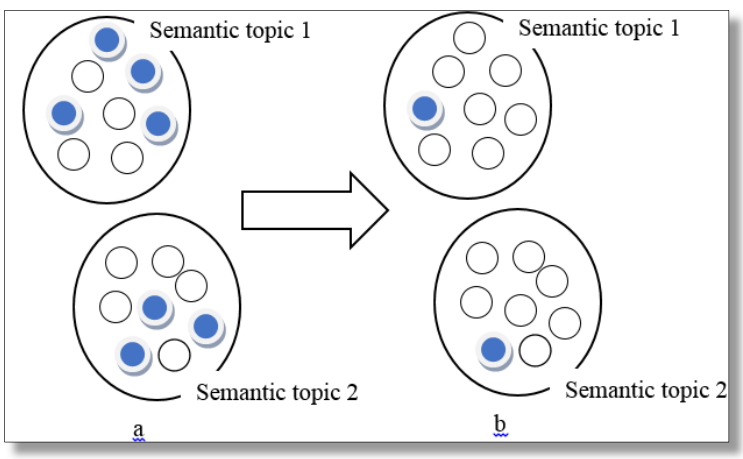


Figure A 2.9 Candidates for semantic term identification (a and b)

Semantic links between semantic terms for the term graph are constructed using the vector measure, one of the measures of relatedness of WordNet::Similarity (Pedersen et al., 2004). The vector measure creates a co–occurrence matrix for each word used in WordNet glosses from a given corpus, and then represents each gloss/concept with a vector that is the average of these co–occurrence vectors.

Let W_i and W_j be semantic terms of the synsets A and B, respectively. Let $\vec{A} = (a_1, \ldots, a_q)$ and $\vec{B} = (b_1, \ldots, b_q)$ be the co–occurrence vectors of A and B, respectively. Let V_z be the set of

semantic terms of the semantic topic Z. The weight of the link between W_i and W_j is computed by equation (A 2.17):

$$Dis(W_i, W_j \mid \mathrm{z}) = \frac{SEM(W_i|\mathrm{z}) + SEM(W_j|\mathrm{z})}{\sum_{W_k \in V_Z} SEM(W_k|\mathrm{z})} \times \sqrt{\sum_{l=1}^{n} (a_l - b_l)^2} \qquad \text{(A 2.17)}$$

To discover a semantic relation between two terms, the semantic distance is computed. The semantic distance between two terms is the shortest path between the terms using equation (A 2.18):

$$SEMDis(W_i, W_j \mid \mathrm{z}) = \underset{pa \in P}{\mathrm{MIN}} \left[\sum_{W_k \in pa} Dis(W_i, W_k \mid \mathrm{z}) \right] \qquad \text{(A 2.18)}$$

where *pa*, W_k, and *P* denote a path between W_i and W_j in the thesaurus, a term on a path *pa* and the set of paths *pa* between W_i and W_j, respectively.

To formally define a semantic relation between two terms W_i and W_j, the semantic distance $SEMDis(W_i, W_j \mid \mathrm{z})$ must not exceed the semantic threshold. The semantic threshold is determined by experimentation.

The last process to generate the semantic term graph BM-SemGraph is a merging of the term graph and the semantic graph. The term graph and semantic graph are merged by coupling the co-occurrence relation and the semantic relation. New terms are added as semantic terms and new links are added as semantic links if they do not appear in the term graph. For each link between two nodes W_j and W_k of the merged graph, the weight, called the BM Weight (BMW), for a given topic t_i is computed using equation (A 2.19):

$$BMW(W_j, W_k \mid t_i) = \frac{\lambda}{SEMDis(W_j, W_k \mid t_i)} + (1 - \lambda) \times w(W_i, W_j) \qquad \text{(A 2.19)}$$

where λ determined by experimentation.

In order to optimize the clusters of BM-SemGraph, the weak links or partitioning of sparse clusters are removed. At this step, each cluster is considered a topic and the terms of the cluster become the terms of the topic.

3.4.3 Topic detection process phase

Figure A 2.10 presents the process used by BM-SATD to assign topics to a document. Topics that may be associated with a new document are detected based on the BM-SemGraph. Note that the BM-SemGraph is obtained using a collection of documents. In this case, the likelihood of detecting topics among a collection of documents is high and must be computed. To accomplish this, the feature vector of each topic based on the clusters of BM-SemGraph is computed. The feature vector of a topic is calculated using the BMRank of each topic term. Let A be the set of nodes of BM-SemGraph directly linked to term W_j in the topic t_i. The score for the term W_j is given by equation (A 2.20):

$$BMRank(W_j|t_i) = \frac{\sum_{W_k \in A} BMW(W_j, W_k \mid t_i)}{|A|} \qquad \text{(A 2.20)}$$

The term with the largest BMRank is called the main term of the topic; other terms are secondary terms. The same processes are used to obtain the BM-SemGraph of an individual document *d* and the feature vectors of topics t_j^d. Next, the similarity between each topic t_i and the topics t_j^d of document *d* is computed in order to detect document topics. Let:

1. W_i be a master term of topics t_j^d and a master or secondary term of t_i;
2. B be the intersection of the set of terms of BM-SemGraph directly linked to term W_j in the cluster of topic t_i and the set of terms of BM-SemGraph of individual document *d* directly linked to term W_j in the cluster of topic t_j^d;
3. C be the union of the set of terms of BM-SemGraph directly linked to term W_j in the cluster of topic t_i and the set of terms of BM-SemGraph of individual document *d* directly linked to term W_j in the cluster of topic t_j^d.

The similarity between t_i and topic t_j^d is computed with equation (A 2.21):

$$Sim(t_i|t_j^d) = \frac{\sqrt{\sum_{W_k \in B} \left(BMW(W_i, W_k \mid t_i) - BMW\left(W_i, W_k \mid t_j^d\right)\right)^2}}{\sqrt{\sum_{W_h \in C} \left(BMW(W_i, W_h \mid t_i) - BMW\left(W_i, W_h \mid t_j^d\right)\right)^2}} \quad \text{(A 2.21)}$$

Here, t_i and topic t_j^d are considered to be similar when their similarity $Sim\left(t_i \middle| t_j^d\right)$ does not exceed the vector similarity threshold.

Finally, the document d is assigned to topics that are similar to its feature vectors. Algorithm 4 of Appendix A gives more detail about the topics detection process for a new document.

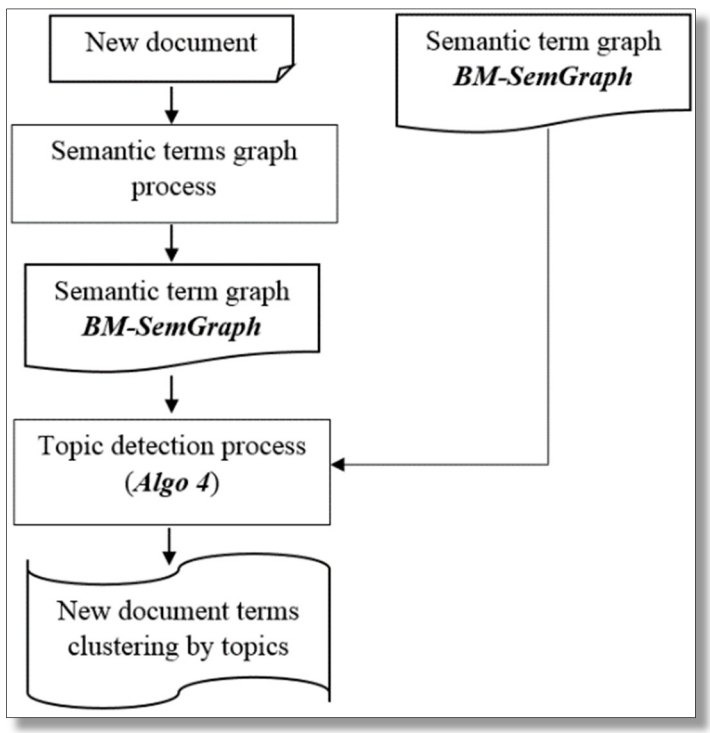


Figure A 2.10 Topic detection process phase - Architecture overview

3.4.4 Training process phase

The training process establishes a terms graph based on the relevant and less similar documents for a given topic t_i. To form the terms graph for a given topic, preprocessing of its relevant and

less similar documents is first carried out, a set of lines is obtained where each line is a list of terms, and the co-occurrence of these terms is then computed.

Let Doc be a document and $V_{Doc} = (w_1, w_2; \ldots, w_N)$ be the terms of Doc. The co-occurrence of $co\left(\overleftrightarrow{W_i, W_j}^{\varepsilon}\right)$ of Wi and W_j where ε denotes the minimum distance between W_i and W_j is computed using equation (A 2.22):

$$co\left(\overleftrightarrow{W_i, W_j}^{\varepsilon}\right) = \sum_{l=1}^{L_{Doc}} \frac{N^{line\ l}(\overleftrightarrow{W_i, W_j}^{\varepsilon})}{\left\lceil \frac{N(line\ l)}{\varepsilon} \right\rceil} \qquad \text{(A 2.22)}$$

where $N^{line\ l}\left(\overleftrightarrow{W_i, W_j}^{\varepsilon}\right)$ denotes the number of times that W_i and W_j co-occur with a minimum distance ε, regardless of the order of appearance, and N(line l) denotes the number of terms of line l.

A relation between two terms W_i and W_j is formally defined when the computed co-occurrence between them exceeds the co-occurrence threshold determined by experimentation. Figure A 2.11 presents an overview of the architecture of the training process phase.

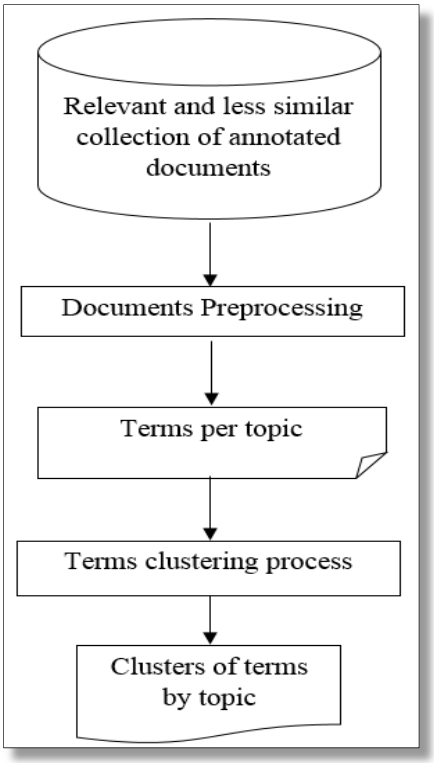


Figure A 2.11 Training process phase - Architecture overview

3.4.5 Topics refining process phase

Figure A 2.12 presents the process used by BM-SATD to refine the detected topics making use of relevant documents already annotated by humans based on existing or known topics. Following this process, three lists of topics are obtained: a list of new topics, a list of similar existing topics and a list of not similar existing topics.

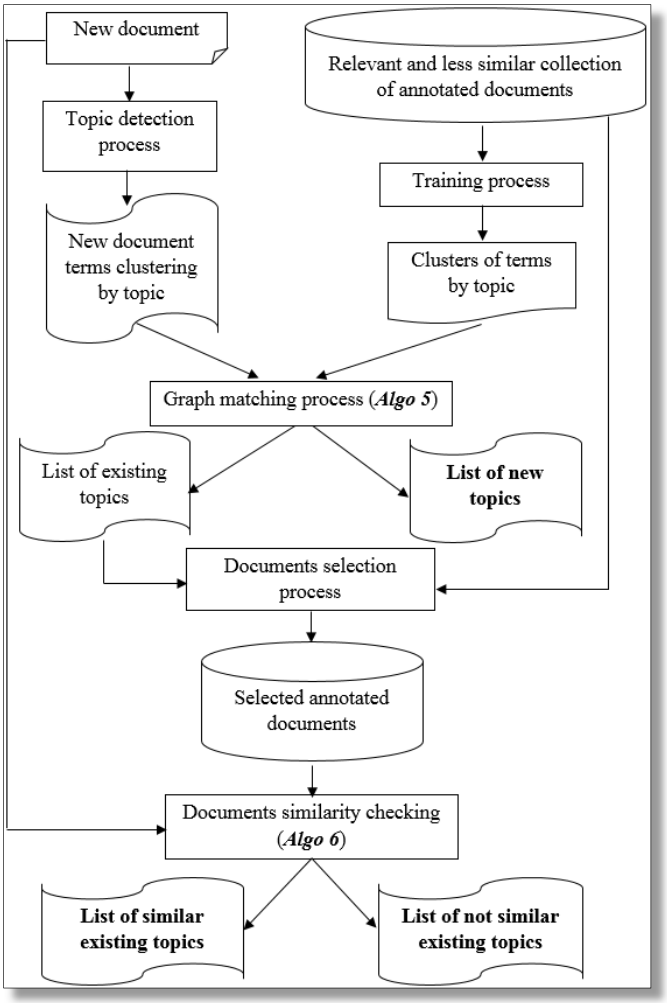


Figure A 2.12 Topic refining process phase - Architecture overview

The list of existing topics that match new document detected topics is identified based on the new document detected topics and annotated documents by topic (existing topics). Then, the clusters of terms by topic (existing topics) are identified based on the collection of relevant and less similar documents. Note that each topic is a cluster of terms graph. Therefore, in this case, a graph matching technique is a good candidate to perform topic similarity detection.

Next, using our graph matching technique, the clusters of terms by topics of relevant and less similar collection of annotated documents which match with CTG are identified, for each cluster of terms graph by topic (CTG) of the new document. The matching score between two clusters is then computed. Let:

1. H be the new document terms graph and G be the terms graph obtained by a training process applied on the collection of relevant and less similar documents annotated by topics;
2. C_j^d be a cluster of H associated to topic t_j^d and C_i be a cluster of G associated with topic t_i;
3. W_i and W_j be two terms of cluster C_j^d; the link matching function g($\overline{W_iW_j}$) between W_i and W_j is defined by equation (A 2.23):

$$g: C_j^d \times C_j^d \rightarrow IR \qquad \text{(A 2.23)}$$

$$g(\overline{W_iW_j}) = \{ \begin{array}{ll} MinHopClusterOfti\,(W_i,W_j) & if\ path\ between\ W_i,W_j \\ 1+MaxHopClusterOfti & if\ not\ path\ between\ W_i,W_j \end{array}$$

For a direct link $\overline{W_iW_j}$ (only one hop between W_i and W_j) of cluster C_j^d, the process checks whether there is a path between W_i and W_j in the cluster C_i, regardless of the number of hops:

1. If paths exist between W_i and W_j in the cluster C_i, $\mathbf{g}(\overline{W_iW_j})$ is the number of hops of the shortest path between W_i and W_j, in term of hops;
2. Otherwise, $\mathbf{g}(\overline{W_iW_j})$ is the number of hops of the longest path that exists in the cluster C_i incremented by 1.

Using the link matching function, the matching score between two clusters C_i^d and C_i is given by equation (A 2.24):

$$o : H \times G \rightarrow]0;1] \qquad \text{(A 2.24)}$$

$$o(C_j^d, C_i) = \frac{|C_j^d|}{\sum_{W_i,W_j \in C_j^d} g(\overline{W_iW_j})}$$

where $|C_j^d|$ is the number of links in clusters C_i^d.

For a better understanding, consider the term graphs in Figure A 2.13.

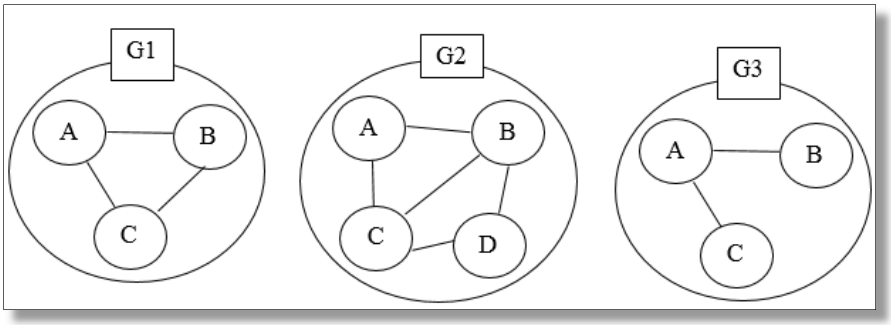


Figure A 2.13 Illustration of term graphs matching score computation

According to Figure A 2.12, o(G1,G2) = 3/3 = 1 while o(G2,G1) = 5/9 and o(G1,G3) = 3/5 while o(G3,G1) = 2/2 = 1. The graph matching technique to identify the existing topics of new document is described by algorithm 5 of appendix A.

The clusters of H and G whose matching scores exceed a term cluster matching threshold are considered as matching and are assumed to be the same topics. Otherwise, the clusters of H that do not match any clusters of G, are assumed to be new topics.

Note that the term cluster matching threshold is determined by experimentation.

Based on the H and G clusters that match, the relevant and less similar documents per existing topic that may have the same topic as the new document are identified. Making use of this set of selected documents, the similarity between the new document and each relevant and less similar document of each existing topic i is measured. Let:

1. D be the union of the new document *d* and a set of relevant and less similar documents of existing topics t_i that are selected by documents selection process;
2. W = {W_1, . . . , W_m} the set of distinct terms occurring in D.

The defined m-dimensional vector represents each document of D. For each term of W, its tf-idf is computed using equation (A 2.1). This allows one to obtain the vector $\overrightarrow{t_d} = (\text{tfidf}(W_1, d, t_i), \ldots, \text{tfidf}(W_m, d, t_i))$. When documents are represented as term vectors, the similarity of two documents corresponds to the correlation between the vectors. Here, cosine similarity is applied to measure this similarity. The cosine similarity is defined as the cosine

of the angle between vectors. An important property of the cosine similarity is its independence of document length.

Given two documents $\overrightarrow{t_{d1}}$ and $\overrightarrow{t_{d2}}$, their cosine similarity is computed using equation (A 2.25):

$$SimCos(\overrightarrow{t_{d1}}, \overrightarrow{t_{d2}}) = \frac{\overrightarrow{t_{d1}} \cdot \overrightarrow{t_{d2}}}{|\overrightarrow{t_{d1}}| \times |\overrightarrow{t_{d2}}|} \quad \text{(A 2.25)}$$

Note that it is already assumed that when the similarity $SimCos(\overrightarrow{t_{d1}}, \overrightarrow{t_{d2}})$ of two documents d1 and d2 is less than the similarity threshold β, the documents are not similar. The computation of document similarity allows BM-SATD to classify the existing topics of new documents into:

1. Similar existing topics,
2. Not similar existing topics.

Details are given in Algorithm 6, Appendix A.

3.5 Semantic sentiment and emotion analysis: BM-SSEA

The aim of BM-SSEA is to classify the corpus of documents taking emotion into consideration, and to determine which sentiment it more likely belongs to.

A document can be a distribution of emotion $p(e|d)\ e \in E$ and a distribution of sentiment $p(s|d)\ s \in S$. BM-SSEA is a hybrid approach that combines a keyword-based approach and a rule-based approach. BM-SSEA is applied at the basic word level and requires an emotional keyword dictionary that has keywords (emotion words) with corresponding emotion labels.

Next, to refine the detection, BM-SSEA develops various rules to identify emotion. Rules are defined using an affective lexicon that contains a list of lexemes annotated with their affect.

The emotional keyword dictionary and the affective lexicon are implemented in a thesaurus. BM-SSEA is a knowledge-based approach that uses an AI computational technique. The

purpose of BM-SSEA is to identify positive and negative opinions and emotions. Figure A 2.14 presents an overview of the architecture of the sentiment and emotion detection process phase.

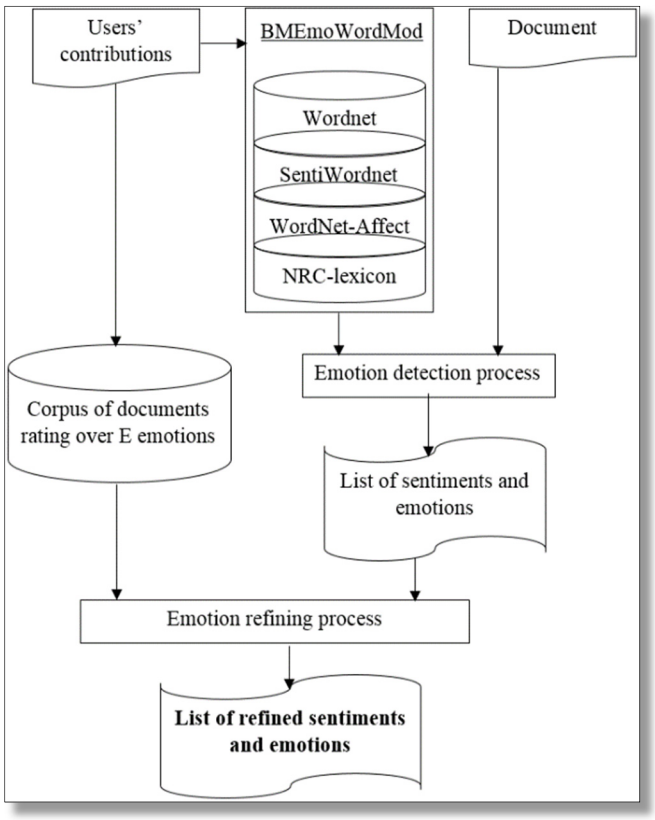


Figure A 2.14 Sentiment and emotion detection process phase – Architecture overview

For affective text evaluation, BM-SSEA uses the SS-Tagger (a part-of-speech tagger) (Tsuruoka & Tsujii, 2005) and the Stanford parser (de Marneffe M-C et al., 2006). The Stanford parser was selected because it is more tolerant of constructions that are not grammatically correct. This is useful for short sentences such as titles. BM-SSEA also uses several lexical resources that create the BM-SSEA knowledge base located in the thesaurus. The lexical resources used are:

1. WordNet,
2. WordNet-Affect,
3. SentiWordNet,

4. NRC emotion lexicon.

WordNet is a semantic lexicon where words are grouped into sets of synonyms, called synsets. In addition, various semantic relations exist between these synsets (for example: hypernymy and hyponymy, antonymy and derivation).

WordNet-Affect is a hierarchy of affective domain labels that can further annotate the synsets representing affective concepts.

SentiWordNet assigns to each synset of WordNet three sentiment scores: positivity, negativity, objectivity, the sum of which always equals 1.0.

The NRC emotion lexicon is a list of English words and their association with eight basic emotions (anger, anticipation, disgust, fear, joy, sadness, surprise and trust) and two sentiments (negative and positive). The NRC emotion lexicon is a thesaurus that associates for a word, the value one or zero for each emotion. This association is made of binary vectors. The disadvantage of this thesaurus is that since the values are binary, all words belonging to an emotion have the same weight for that emotion. To address this problem, the NRC emotion lexicon thesaurus was combined with the WordNet, WordNet-Affect and SentiWordNet thesaurus. This associates a feelings score with each word-POS. POS_1 are grammatical categories used to classify words in dimensions such as adjectives or verbs. SentiWordNet associates with each couple a valence score that can be either negative or positive with respect to the sense of the word in question. The word death, for example, is likely to have a negative score. BM-SSEA also relies on shifter valences. These are lexical expressions capable of changing the valence score of emotions in a text.

For example, take the phrase ″I am happy″ with a score of 1 for the joy emotion. For the phrase "I am **very** happy", ‘very’ is a valence intensifier that will change the joy emotion score to 2. In the case, "I am **not** happy" the modifier ‘not’ will change the emotion joy to the contrary emotion sadness.

The main *component* of BM-SSEA is the thesaurus, called BM emotion word model (BMEmoWordMod). BMEmoWordMod is an emotion-topic model that provides the emotional score of each keyword by taking the topic into account.

BMEmoWordMod introduces an additional layer (i.e., latent topic) into the emotion-term model such as SentiWordNet. BM-SSEA is composed of three phases:

1. BMEmoWordMod generation process phase,
2. Sentiment and emotion discovery process phase,
3. Sentiment and emotion refining process phase.

The following sub-sections describe the three phases of the BM-SSEA model used to discover sentiment and emotion.

3.5.1 BMEmoWordMod generation process phase

In the first step, a training set from the original corpus is created. The most relevant and discriminative documents are selected automatically. In the second step, each word is tagged with a POS and the combination of word and POS used as the essential feature. Finally, BMEmoWordMod is generated using the extracted features, which can then be used to discovery the sentiments and emotions of new documents.

Basically, a BMEmoWordMod entry has the following fields <Word/POS/synsets_ID><Topics><Emotion_Probability><Sentiment_Probability> where:

1. Emotion_Probability is a vector of ordered emotion label probability such as <anger probability, disgust probability, fear probability, joy probability, sadness probability, surprise probability>;
2. Sentiment_ Probability is a vector of ordered sentiment category probability such as <positive score, negative score>.

For example, the BMEmoWordMod entry for “kill” may look like: <kill/v/00829041><War><0.5, 0.1, 0.3, 0, 0.2, 0><0.1, 0.6>.

Step 1: Training set selection

The objective of this step is to reduce the time for generating the emotion lexicon BMEmoWordMod, while obtaining a better quality lexicon. For each emotion e_i, documents in the corpus are ranked by descending order of ratings over e_i. Next, the emotions with the highest ratings among the documents are chosen. Then relevant documents for a given emotion e_i are selected using the first phase of BM-SATD (see section 3.4.1 of BM-SATD). The training set selection process terminates when the first phase BM-SATD requirements are meet. The training set TS is produced by conducting this step on the entire corpus.

Step 2: Intermediate lexicon generation

Using WordNet-Affect, the WordNet entries are filtered in order to retain only those synsets where the A_labeb is "EMOTION". Then, using SentiWordNet and the NRC emotion lexicon, the sentiment category and emotion value are associated with each selected emotional synset of WordNet. An intermediate lexicon is produced where each entry is <word/POS/synsets_ID><Emotion_value><Sentiment_Score>.

BMEmoWordMod evaluates the probability of each emotion based on the topic and user rating.

Step 3: Sentiment and emotion lexicon generation

The assumption that words in a document are the first indicator of the evoked emotion is assumed to be valid. However, the same word in different contexts may reflect different emotions, and words that bear emotional ambiguity are difficult to recognize out of context. Thus, other strategies are necessary to associate a sentiment or emotion with a given word. The POS of each word is used to alleviate the problem of emotional ambiguity of words and the context dependence of sentiment orientations. The POS of a word is a linguistic category defined by its syntactic or morphological behaviour. Categories include: noun, verb, adjective, adverb, pronoun, preposition, conjunction and interjection.

For example, the word ‘‘bear’’ has completely different orientations, one positive and one negative, in the following two sentences:

1. Teddy bear: a helping hand for disease sufferers;
2. They have to bear living with a disease.

The word ‘‘bear’’ is a noun in the first sentence and a verb in the second. A word feature f_j is defined as the association of the word W_j and its POS, e.g., (Kill/Verb). After defining the word feature f_j, its emotion probability is computed with equation (A 2.26):

$$\mathrm{EmoPro}(e_i|f_j,t_k) = Val(f_j) \times \frac{\sum_{\mathrm{d} \in C_{tk} \subset \mathrm{ND}} p(f_j,t_k,d) \times oc(e_i,t_k)}{\sum_{e_l \in E} \sum_{\mathrm{d} \in C_{tk} \subset \mathrm{ND}} p(f_j,t_k,d) \times oc(e_l,t_k)} \quad \text{(A 2.26)}$$

where $Val(f_j)$ denotes the value (1 or 0) of word feature f_j in the intermediate lexicon, and where:

1. $p(f_j,t_k,d)$ denotes the probability of feature f_j conditioned on document of corpus C_{tk} (subset of documents with topic t_k);
2. $p(f_j,t_k,d)$ is the number of occurrences of the feature f_j in d divided by the total number of occurrences of all features in d;
3. $oc(e_i,t_k)$ denotes the co-occurrence number of documents d of C_{tk} and emotion e_i.

This strategy is used to eliminate emotions that are not associated with the same word in the NRC emotion lexicon. The sentiment probability of the word feature f_j is given by equation (A 2.27):

$$\mathrm{SenPro}(s_i|f_j,t_k) = SSco(f_j) \times \frac{\sum_{\mathrm{d} \in C_{tk} \subset \mathrm{ND}} p(f_j,t_k,d) \times oc(s_i,t_k)}{\sum_{s_l \in S} \sum_{\mathrm{d} \in C_{tk} \subset \mathrm{ND}} p(f_j,t_k,d) \times oc(s_l,t_k)} \quad \text{(A 2.27)}$$

where:

1. $SSco(f_j)$ denotes the score of feature f_j in the intermediate lexicon.

2. $p(f_j, t_k, d)$ denotes the probability of feature f_j conditioned on the document of corpus C_{tk} (sub set of documents with topic t_k).
3. $oc(s_i, t_k)$ denotes the co-occurrence number of documents d of C_{tk} and sentiment s_i.

Here, s_i may have two values, a positive sentiment S_P and negative sentiment S_N. Finally, to derive BMEmoWordMod, first the topic is added, then the emotion value is replaced by the computed emotion probability and the sentiment score with the computed sentiment probability.

3.5.2 Sentiment and emotion discovery process phase

This phase identifies the sentiments and emotions that are likely associated with a given new document by using the sentiment and emotion semantic lexicon BMEmoWordMod generated in the previous section. After preprocessing, the term vector of the new document is defined using TF-IDF.

Let ND be the new document and $W_{ND} = \{W_1, \ldots, W_z\}$ the set of distinct terms occurring in the corpus of documents. To obtain the z-dimensional term vector that represents each document in the corpus, the tf-idf of each term of W_z is computed. The result of this computation establishes the term vector $\overrightarrow{t_{ND}} = (\text{tfidf}(W_1, ND), \ldots, \text{tfidf}(W_z, ND))$.

Using vector $\overrightarrow{t_{ND}}$, $T_{ND}=\{t_p, \ldots, t_q\}$ obtained using BM-SATD and BMEmoWordMod, the sentiment and emotion vector of new document $\overrightarrow{E_{f_j,ND}} = (\mathrm{E}(f_j, \mathrm{ND}, e_1), \ldots, \mathrm{E}(f_j, \mathrm{ND}, \mathrm{e_E}), \mathrm{E}(f_j, \mathrm{ND}, s_\mathrm{P}), \mathrm{E}(f_j, \mathrm{ND}, s_\mathrm{N}))$ is given by equation (A 2.28):

$$\mathrm{E}(f_j, \mathrm{ND}, e_i) = \frac{\text{tfidf}(W_j, ND)}{\sum_{l=1}^{z} \text{tfidf}(W_l, ND)} \times \sum_{\mathrm{t_k} \in \mathrm{T_{ND}}} \text{BMEmoWord}(f_j, e_i, \mathrm{t_k}) \qquad \text{(A 2.28)}$$

where $\text{BMEmoWord}(f_j, \mathrm{e_i}, \mathrm{t_k})$ denotes the emotion probability of emotion ei for the feature word fj giving the topic tk. $\text{BMEmoWord}(f_j, \mathrm{e_i}, \mathrm{t_k})$ is selected in BMEmoWordMod.

The weight of emotion ei for document ND is computed with equation (A 2.29):

$$\mathrm{W_E}(\mathrm{ND}, e_i) = \sum_{\mathrm{W_j} \in \mathrm{W_{ND}}} \mathrm{E}(f_j, \mathrm{ND}, e_i)) \tag{A 2.29}$$

Equation (A 2.29) yields the emotional vector of new document ND.

$$\overrightarrow{V_{ND}} = \left(\mathrm{W_E}(\mathrm{ND}, e_1), \dots, \mathrm{W_E}(\mathrm{ND}, e_i), \dots, \mathrm{W_E}(\mathrm{ND}, e_E), \mathrm{W_E}(\mathrm{ND}, s_P)\right), \mathrm{W_E}(\mathrm{ND}, s_N))).$$

Next, the new document ND emotion and sentiment is inferred using a fuzzy logic approach and the emotional vector $\overrightarrow{V_{ND}}$. The weight of emotion is transformed into five linguistic variables: very low, low, medium, high, and very high. Then, using these variables as input to the fuzzy inference system one obtains the final emotion for the new document. The fuzzy logic rules are predefined by experts.

3.5.3 Sentiment and emotion refining process phase

The refining process validates discovered sentiment and emotion after the document analysis. Similarity is computed between new documents and documents in the corpus rated over E emotions. First, the term vectors of each document are defined using the tf-itf of each term, tf-itf is then computed using equation (A 2.1). Note that the terms extracted from the corpus of documents rated over E emotions are those employed by users.

Next, to measure the similarity between two documents, the cosine similarity of their representative vectors is computed using equation (A 2.25) and algorithm 6. Two documents d1 and d2 are similar when the similarity $SimCos(\overrightarrow{t_{d1}}, \overrightarrow{t_{d2}})$ of these two documents is less than the similarity threshold β.

4. Evaluation using simulations

This section presents an evaluation of BM-SATD and BM-SSEA performance using simulations. To perform these simulations, an experimental environment called Libër was

used. Libër was developed to provide a simulator to prototype the different algorithms of SMESE V3.

4.1 Dataset and parameters

To evaluate BM-SATD and BM-SSEA, real datasets from different projects that have digital and physical library catalogues were used. These datasets, consisting of 25,000 documents with a vocabulary of 375,000 words, were selected using average TF-IDF for the analysis. The documents covered 20 topics and 8 emotions. The number of documents per topic or emotion was approximately equal. The average number of topics per document was 7 while the average rating emotion number per document was 4. 15,000 documents of the dataset were used for the training phase and the remaining 100 used for the test. Note that the 10,000 documents used for the tests were those that had more annotated topics or a higher rating over emotions.

To measure the performance of topic detection (sentiment and emotion discovery, respectively) approaches, comparison of detected topics (the discovered sentiment and emotion, respectively) with annotation topics of librarian experts (user ratings) were carried out. Table A 2.2 presents the values of the parameters used in the simulations. The server characteristics for the simulations were: Dell Inc. PowerEdge R630 with 96 Ghz (4 x Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz, 10 core and 20 threads per CPU) and 256 GB memory running VMWare ESXi 6.0.

Table A 2.2 Simulation parameters

Parameter	Value	Parameter	Value
ε	3	α	100
NumKeyTerm	8	co-occurrence threshold	0.75
ω	0.5	semantic threshold	1
β	0.7	term cluster matching threshold	0.45
λ	0.6		

4.2 Performance criteria

BM-SATD and BM-SSEA performance was measured in terms of running time (P. Chen et al., 2016) and accuracy (C. Zhang et al., 2016) (Sayyadi & Raschid, 2013). Note that in the library domain, the most important criteria was precision while resource consumption was important for the software providers.

The running time, denoted by Rt, was computed as follows:

$$Rt = Et - Bt$$

where Et and denotes the time when processing is completed and Bt the time when it started.

To compute the accuracy, let Tannotated and Tdetected be the set of annotated topic and the set of detected topics by BM-SATD for a given document d. The accuracy of topics detection, denoted by A_d^t, was computed as follows:

$$A_d^t = \frac{2.|T_{\text{annotated}} \cap T_{\text{detected}}|}{|T_{\text{annotated}}| + |T_{\text{detected}}|}$$

The same formula was applied to compute the accuracy of the sentiment and emotion discovery measurement. Erating (resp. Ediscovered) that denotes the set of rating over emotion (resp. the set of discovered emotion by BM-SSEA) was used instead of Tannotated (resp. Tdetected).

Simulation results were averaged over multiple runs with different pseudorandom number generator seeds. The average accuracy, Ave_acc, of multiple runs was given by:

$$Ave\_acc = \frac{\sum_{x=1}^{I}\left(\frac{\sum_{d \in TD} A_d^t}{|TD|}\right)}{I}$$

where TD denotes the number of tests documents and I denotes the number of test iterations.

The average running time, Ave_run_time, was given by:

$$Ave\_run\_time = \frac{\sum_{x=1}^{I} Rt}{I}$$

4.3 Topic detection approaches performance evaluation

BM-SATD performance was evaluated in terms of running time and accuracy. The dataset and parameters mentioned above were applied. BM-SATD performance was compared to the approaches described in (C. Zhang et al., 2016), (Sayyadi & Raschid, 2013), (David M. Blei et al., 2003) and (P. Chen et al., 2016), referred to as LDA-IG (probabilistic and graph approach), KeyGraph (graph analytical approach), LDA (probabilistic approach) and HLTM (probabilistic and graph approach), respectively. LDA-IG, KeyGraph, LDA and HLTM were selected because they are text-based and long text approaches.

4.3.1 Comparison approaches

Table A 2.3 presents the characteristics of the comparison approaches. Our approach BM-SATD is the only one that is really semantic and takes into account the correlated topic and domain knowledge. The parameters for the comparison approaches used where those which provided the best performance.

Table A 2.3 Topic detection approaches for comparison

Approach	Granu larity	Description	Training phase	Refining	Semantic	Topic corre-lation	Domain knowledge
LDA-IG (C. Zhang et al., 2016)	Document	Probabilistic and graph based	Yes	No	No	No	No
KeyGraph (Sayyadi & Raschid, 2013)	Document	Graph based	Yes	No	No	No	No
LDA (David M. Blei et al., 2003)	Document	Probabilistic based	No	No	No	No	No
HLTM (P. Chen et al., 2016)	Document	Probabilistic and graph based	Yes	No	No	No	No
BM-SATD	Configu-rable as desired	Semantic, probabilistic and graph based	Yes	Yes	Yes	Yes	Yes

4.3.2 Results analysis

Figure A 2.15 presents the average running time of the detection phase when the number of documents used for the tests were varied. Training times were excluded as this phase was performed only one time. However, the BM-SATD training phase required more time than the other approaches. This was justified by the fact that BM-SATD identifies the relevant and less similar documents used for training phase. Fortunately, the new generation of data center equipment offers sufficient resources to reduce significantly the training delay. Thus, only the time required to detect new document topics (subject) was measured.

Figure A 2.15 also shows that the average running time increased with the number of test documents. Indeed, the bigger the number of test documents, the longer the time to perform detection and, ultimately, the higher the average running time.

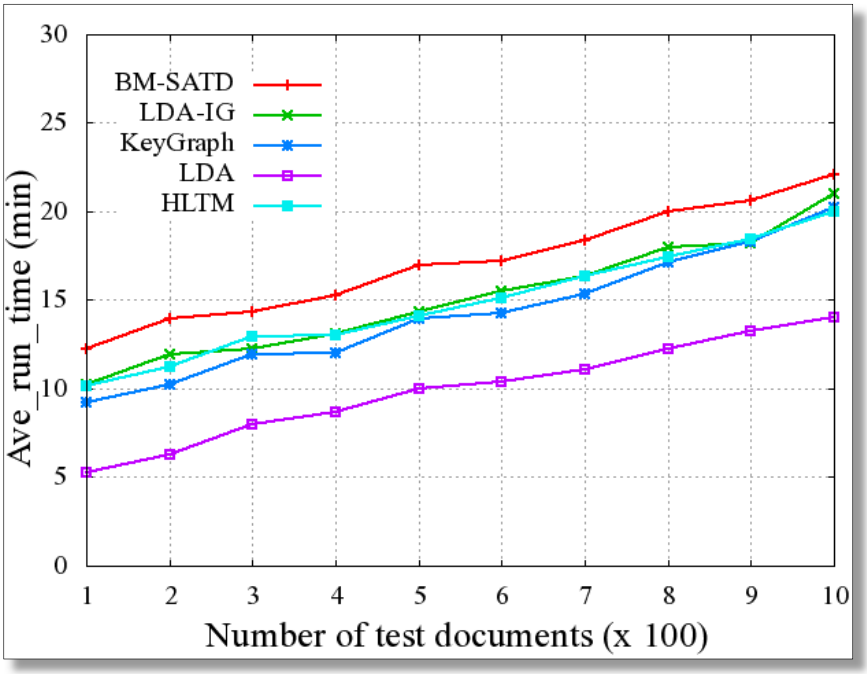


Figure A 2.15 Topic detection - Average running time versus number of documents for test phase

It was also observed that LDA outperforms the other approaches. LDA produced an average of 1.37 sec per document whereas BM-SATD produced an average of 2.62 sec per document.

The average relative improvement (defined as [Aver._runtime of BM-SATD – Aver._runtime of LDA]) of LDA compared with BM-SATD was approximately 1.25 sec per document. The short run times of LDA were due to the fact that LDA did not perform a graph treatment. Graph processing algorithms are very time consuming. Other approaches also outperformed BM-SATD on the running time criteria since BM-SATD performed topic refining in order to increase accuracy.

Figure A 2.16 shows the average accuracy when varying the number of detected topics. For the five approaches, the average accuracy decreased with the number of detected topics. The

increase in the number of subjects to detect led to decreased accuracy. However, in terms of accuracy, BM-SATD outperformed the approaches used for comparison. BM-SATD produced an average accuracy of 79.50% per topic while LDA-IG, the best among the approaches used for comparison, produced an average of 61.01% per topic.

The average relative improvement in accuracy (defined as [Ave_acc of BM-SATD – Ave_acc of LDA-IG]) of BM-SATD compared to LDA-IG was 18.49% per topic. The performance of BM-SATD is explained as follows:

1. BM-SATD used the relevant documents for training phase;
2. BM-SATD refined its detection topic results by measuring new document similarity with relevant and less similar annotated documents;
3. BM-SATD combined correlated topic model and domain knowledge model instead of LDA.

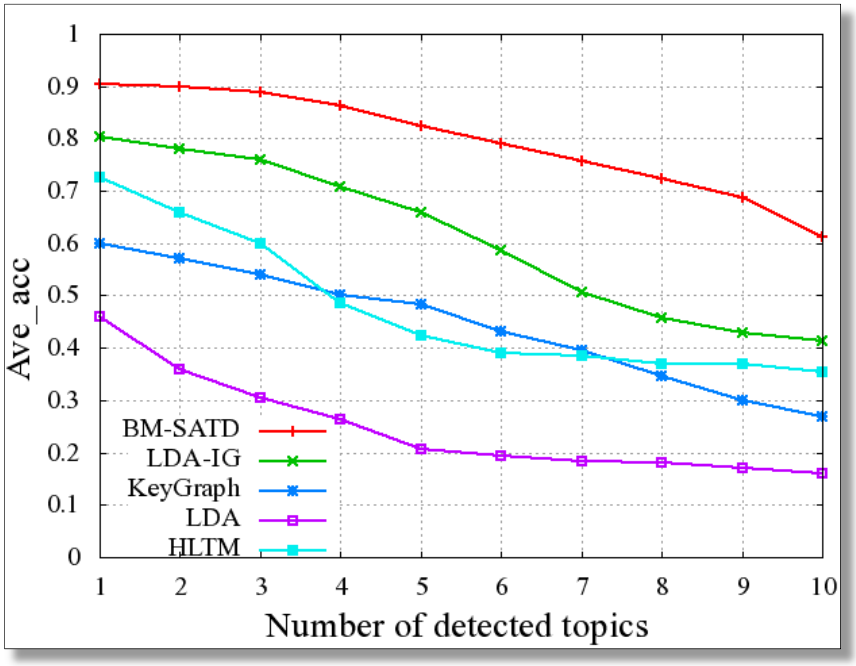


Figure A 2.16 Accuracy for number of detected topics for 5 comparison approaches

Figure A 2.16 also shows that BM-SATD produced an average accuracy of 90.32% for one detected topic and 61.27% for ten detected topics compared to 80.29% and 41.01%

respectively for LDA-IG. The gap between BM-SATD accuracy and LDA-IG accuracy was 10.03% for one detected topic and 20.26% for ten detected topics. This meant that BM-SATD was by in large more accurate than LDA-IG in detecting several topics.

The Figure A 2.17 presents the average accuracy when varying the number of training documents of the learning phase. LDA was not included in the scenario since no training phase was performed. Figure A 2.17 shows that the average accuracy increased with the number of training documents. The larger the number of training documents, the better the knowledge about word distribution and co-occurrence and, ultimately, the higher the detection accuracy. However, the accuracy remained largely stable for very high numbers of training documents. When the number of documents of a collection was larger, the number of vocabulary words remained constant, and the term graph did not change. It also shows that HLTM was the approach whose detection accuracy was the first to reach stability at 10,000 training documents. HLTM builds a tree instead of a graph as the other approaches and its tree has less internal roots to identify topics. However, BM-SATD and LDA-IG outperformed HLTM in terms of accuracy.

Figure A 2.17 also shows that BM-SATD outperformed LDA-IG on the accuracy criteria. For example, BM-SATD demonstrated an average accuracy of 73.49% per 2,000 training documents while LDA-IG produced an average accuracy of 50.86% per 2,000 training documents. The average relative improvement of BM-SATD compared to LDA-IG was 22.63% per 2,000 training documents. The better performance of BM-SATD followed from its use of a domain knowledge model. BM-SATD did not require a large number of documents for the training phase.

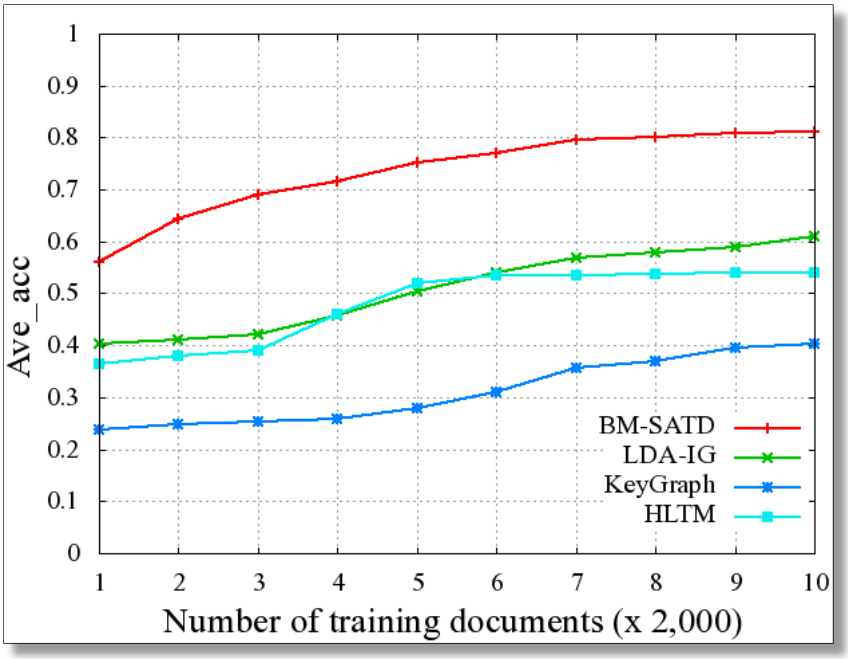


Figure A 2.17 Topic detection - accuracy for number of training documents

In conclusion, the 1.25 sec running time per document increase was a small price to pay for the larger average accuracy of topic detection (18.49%).

4.4 Sentiment and emotion analysis performance evaluation

BM-SSEA performance was also evaluated in terms of accuracy and running time. Simulations used the dataset and parameters mentioned previously. The performance of BM-SSEA was compared to the approaches described in (Bao et al., 2012) and (Anusha & Sandhya, 2015), referred to as ETM-LDA and AP, respectively. ETM-LDA and AP were selected because they were document-based rather than phrase-based.

4.4.1 Comparison of approaches with BM-SSEA

Table A 2.4 shows the characteristics of the approaches used for comparison with BM-SSEA. BM-SSEA was the only entirely semantic approach taking into account the rules for inferring emotion. In addition, BM-SSEA used a semantic lexicon. Several approaches used semantic

lexicon, but these were limited to phrases rather than documents. The best performance approaches used were AP and ETM_LDA.

Table A 2.4 Sentiment and emotion approaches for comparison

Approach	Granularity	Approach	Training phase	Refining	Thesaurus	Topic modeling	Emotions number
AP (Anusha & Sandhya, 2015)	Document	Learning based	Yes	No	5	No	8
ETM-LDA (Bao et al., 2012)	Document	Keyword based	Yes	No	6	Yes	8
BM-SSEA	Configurable as desired	Keyword and rule based	Yes	Yes	1, 2, 3, and 4	Yes	8

1-WordNet; 2-WordNet-Affect; 3-SentiWordNet; 4-NRC Emotion Lexicon; 5- Stanford CoreNLP; 6-Gibbs sampling.

4.4.2 Results analysis

Figure A 2.18 presents the average running time when varying the number of detected emotions. As in Figure A 2.17, training times were excluded because this phase was performed only once.

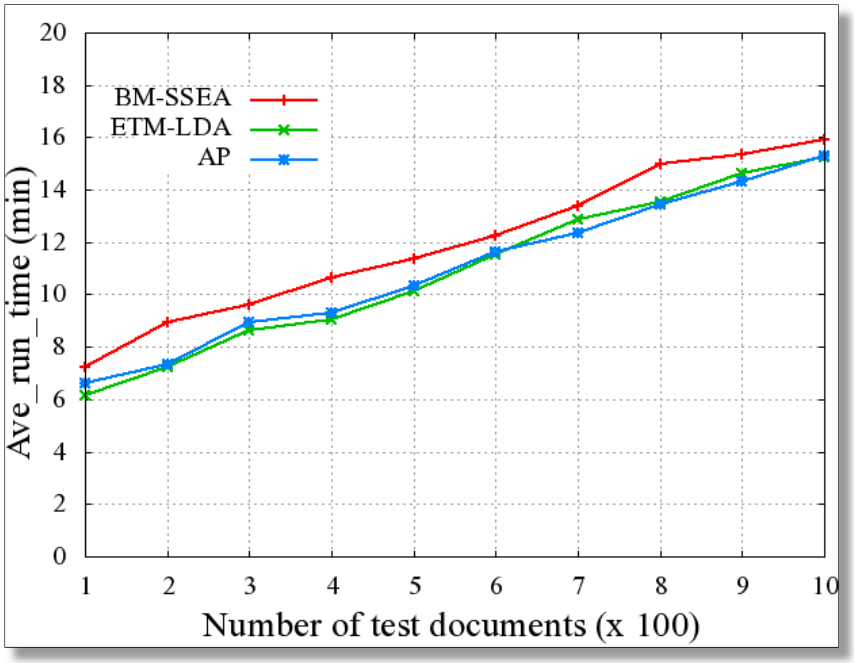


Figure A 2.18 Emotion discovery - Average running time versus number of documents for test phase

The BM-SSEA training phase took more time than the other approaches due to lexicon aggregation and enrichment by users. The average running time increased with the number of test documents. This is normal, as the larger the number of test documents the longer the average running time to perform the sentiment and emotion discovery. Figure A 2.18 shows that ETM-LDA and AP outperformed BM-SSEA on the running time criteria. ETM-LDA required an average of 1.53 sec per document whereas BM-SSEA required an average of 1.74 sec per document. The average relative improvement of ETM-LDA compared with BM-SSEA was approximately 0.21 sec per document. The poorer performance of BM-SSEA resulted from refining sentiment and emotion to increase accuracy.

Figure A 2.19 presents the average accuracy when varying the number of discovered emotions. Positive and negative sentiments were not considered in the accuracy measurement. Figure A 2.19 also shows that the average accuracy decreased with the number of discovered emotions. However, BM-SSEA outperformed the other two approaches used for comparisons. BM-SSEA demonstrated an average accuracy of 93.30% per emotion while ETM-LDA, the best of the other two approaches used for comparison, produced 68.65% accuracy per emotion. The

average relative improvement in accuracy of BM-SSEA compared to ETM-LDA was 24.65% per emotion.

In conclusion, the 0.21 sec running time per document increase was, again, a small price to pay for the larger average accuracy of emotion discovery (24.65%).

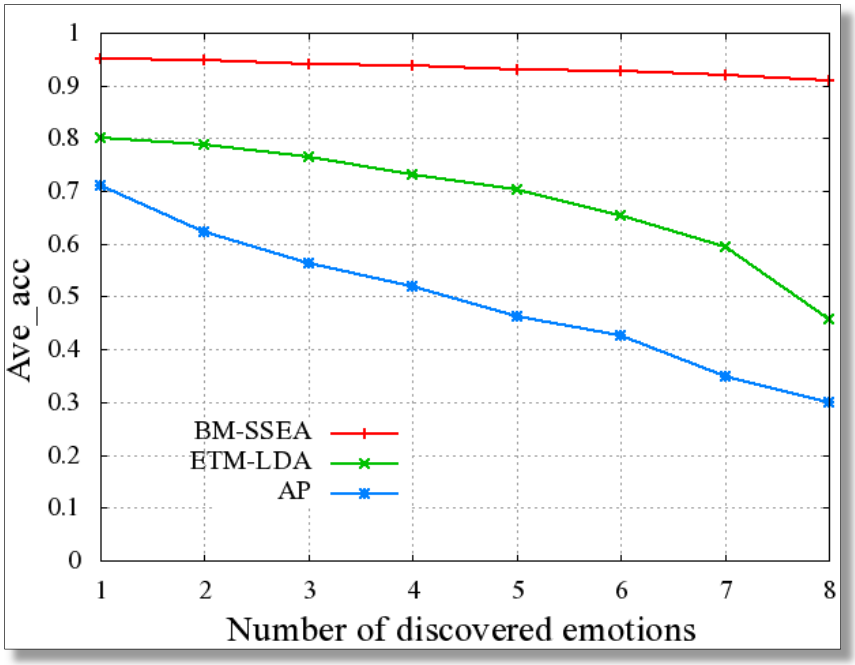


Figure A 2.19 Average detection accuracy for the number of discovered emotions

5. Summary and future work

In this paper, the goal was to increase the findability (search, discover) of entities based on user interest using external and internal semantic metadata enrichment algorithms. As computers struggle to understand the meaning of natural language, enriching entities semantically with meaningful metadata can improve search engine capability. Words themselves have a wide variety of definitions and interpretations and are often utilized inconsistently. While topics, sentiments and emotions may have no relationship to individual words, thesauri express associative relationships between words, ontologies, entities and a

multitude of relationships represented as triplets. From these relationships and defined entities it was possible to dynamically build up a large semantic metadata master catalogue (SMMC).

This paper presented an enhanced implementation of SMESE using metadata and data from the linked open data, structured data, metadata initiatives, concordance rules and authority's metadata to create the SMMC. SMMC offers a foundation for an entire interest-based digital library of semantic mining activities, such as search, discovery and interest-based notifications. Finding bibliographic references or semantic relationships in texts makes it possible to localize specific text segments using ontologies to enrich a set of semantic metadata related to topic or sentiment and emotion.

To help users find interest-based contents, this paper proposes to enhance the SMESE platform through text analysis approaches for sentiments and emotions detection. SMESE V3 can be used (or: makes it possible) to create a semantic master catalogue with enriched metadata that enables search and discovery interest-based processes. This paper presents the design, implementation and evaluation of a SMESE V3 platform using metadata and data from the web, linked open data, harvesting and concordance rules, and bibliographic record authorities. The SMESE includes three distinct processes that:

1. Discover enriched sentiment and emotion metadata hidden within the text or linked to multimedia structure using the proposed BM-SSEA (BM-Semantic Sentiment and Emotion Analysis) algorithm;
2. Implement rule-based semantic metadata internal enrichment (RSMIEE includes algorithms BM-SATD and BM-SSEA);
3. Generate semantic topics by text, and multimedia content analysis using the proposed BM-SATD (BM-Scalable Annotation-based Topic Detection) algorithm.

Furthermore, SMESE V3 provides:

1. An enhanced semantic metadata meta-catalogue (SMM),
2. An enhanced harvesting metadata & data and OpenURL.

The semantic aggregation of metadata content repository offers a foundation for an interest-based digital library of semantic mining activities, such as search, discover and smart notifications.

Table A 2.1 shows the comparison with most known text mining algorithms (e.g., AlchemyAPI, DBpedia, Wikimeta, Open Calais, Bitext, AIDA, TextRazor) and a new algorithm SMESE with many attributes including keyword extraction, classification, sentiment analysis, emotion analysis, and concept extraction. It was noted that SMESE algorithms support more attributes than any other algorithms.

In future work, the focus will be to generate learning-based literature review enrichment and abstract of abstract. STELLAR (Semantic Topics Ecosystem Learning-based Literature Assisted Review) assess each citation to determine her ranking and her inclusion in the final literature assisted review (LAR). One goal of this ecosystem is to reduce reading load by helping researcher to read only the intelligent selection of documents. Using text data mining, machine learning, and a classification model that learn from users annotated data and detected metadata.

Appendix A: BM-SATD Processes, Phases and Algorithms

1. Relevant and less similar document selection phase

This phase identifies the corpus of relevant and similar documents for a given topic. Three algorithms are defined and described in the following steps.

Step 1: Selection of representative documents of a given corpus by topic

In this step, the most relevant documents of each topic are selected. The objective is to reduce the number of documents that used to compute the similarity with a new document in order to detect its topics. Each document of a topic is checked as to whether or not its most important terms are the same as the most important terms of the topic.

Let $C_{ti}=\{D_1,\ldots,D_j,\ldots, D_{Mi}\}$ be the corpus of documents with t_i as topic and $V_{ti}=\{W_1,\ldots,W_k,\ldots, W_{Ni}\}$ be the vocabulary of the topic t_i where each element of V_{ti} is in at least one document of corpus C_{ti}.

Let $D_j=\{W_a\ldots,W_i,\ldots, W_{|Dj|}\}$ be the set of words of document D_j. To obtain D_j, the preprocessing phase is performed which consists of the following processes:

1. Segmentation is a process of dividing a given document into sentences;
2. Stop words are removed from the text. Stop words are frequently occurring words such as 'a', 'an', 'the' that provide less meaning and contain noise. Stop words are predefined and stored in an array;
3. Tokenization separates the input text into separate tokens. Punctuation marks, spaces and word terminators are word breaking characters;
4. Word stemming converts each word into its root form by removing its prefix and suffix for comparison with other words.

The algorithm of step 1 is the following (**<u>Algorithm 1</u>**):

1. For each topic t_i of T
   a) For each D_j of C_{ti}
      - For each W_i of D_j
        - Compute TF-IDF of W_i in the corpus of documents C_{ti} with the following formula:

$$f(W_i, D_j, C_{ti}) = TF - IDF(W_i, D_j, C_{ti}) = TF(W_i, D_j) * \log(\frac{|C_{ti}| = M_i}{IDF(W_i, C_{ti})})$$

where $TF(W_i, D_j)$, $IDF(W_i, C_{ti})$ and M_i denote the number of occurrences of W_i in document D_j, the number of documents in the corpus C_{ti} where W_i appears, and the number of documents in the corpus C_{ti}, respectively.

At this level, for each document D_j of C_{ti}, the set of vectors $V_{Dj}= \{ (W_a, f(W_a,D_j,C_{ti})), \ldots, (W_i, f(W_i,D_j,C_{ti})) ,\ldots, (W_{|Dj|}, f(W_{|Dj|},D_j,C_{ti}))\}$ is obtained where in the couple $(W_i, f(W_i,D_j,C_{ti}))$:

1. W_i denotes a term,
2. $f(W_i,D_j,C_{ti}))$ is its tf-idf *within* the whole corpus C_{ti}.

2. For each topic t_i of T
   a) For each W_k of V_{ti}
      - Compute TF-ITF of W_k for the whole corpus of documents with the following formula:

$$g(W_k, t_i) = TF - ITF(W_k, t_i) = TF(W_k, t_i) * \log(\frac{|T| = n}{ITF(W_k)})$$

where $TF(W_k, t_i)$, $ITF(W_k)$ and |T| denote the number of occurrences of W_k in all the documents of corpus C_{ti}, the number of topics where W_k appears, and the number of topics, respectively.

At this level, for each topic t_i of T, the set of vectors V_{ti}= { $(W_1, g(W_1,t_i)), \ldots, (W_k, g(W_k,t_i)), \ldots, (W_{Ni}, g(W_{Ni},t_i))$} is obtained where in the couple $(W_k, g(W_k,t_i))$, W_k denotes a term and $g(W_k,t_i)$ is its tf-itf in the whole corpus T.

At this stage, the standard deviation σ and the average *avg* number of distinct terms in the documents for the topic is computed in order to decide the number of terms to consider whether the document is relevant to the topic or not. Standard deviation gives a good indication of the dispersion of data to the average.

3. For each topic t_i of T
   a) Compute *avg* of t_i as avg_{ti}
      - $avg_{ti} = \frac{\sum_{j=1}^{|C_{ti}|=M_i} |D_j|}{|C_{ti}|=M_i}$
   b) Compute σ of t_i as σ_{ti}
      - $\sigma_{ti} = \sqrt{\frac{\sum_{j=1}^{|C_{ti}|=M_i} (|D_j| - avg_{ti})^2}{|C_{ti}|=M_i}}$
   c) Compute the number of distinct terms to consider with the following formula:
      $E_{ti} = avg_{ti} - \sigma_{ti}$

   E_{ti} represents approximately 75% of term distribution number per document D_j of C_{ti}.

The score of each document Dj in the topic t_i is then computed as follows:

4. For each topic t_i of T
   a) For each D_j of C_{ti}
      - Classify the terms of D_j using TF-IDF in descending order.
      - $BMscore\left(D_j\right) = \sum_{|E_i|} g(W_i, t_i)$

        where $\sum_{|Ei|}$ are the first $|E_i|$ terms of D_j with the highest tf-idf in the whole corpus C_{ti}..
   b) The **α** documents with the highest BMscore that form the set of documents contained in the relevant documents of topic t_i is selected. Note that **α** is a threshold to be defined.

$$C_{ti} = \left[C'_{ti} = \bigcup_{\alpha}\{D_k\}\right] \cup \left[\bigcup_{M_i-\alpha}\{D_j\}\right] \;/\; \text{ with BMScore}(D_k) > BMScore(D_j)$$

$C'_{ti} = \{D_{k_1}, \dots, D_{k_i}, \dots, D_{k_\alpha}\}$ where $M_i > M'_i = \alpha$ is obtained.

Step 2: Selection of less similar documents of a given corpus by topic

The objective of this step is to retain documents that are less similar among the relevant documents of a given topic t_i C'_{ti}. This avoids having to consider too similar documents in the same topic set and increases the accuracy of detecting a topic in a new document.

- Let C'_{ti} be relevant documents of a given topic t_i. Notice that the documents of C'_{ti} are ordered based on their BMscore.
- Let ***β*** be a similarity threshold. ***β*** is a threshold defined through empirical experimentation.
- Let $C''_{ti} = \{D_{k_1}\}$, where D_{k_1} is the document of C'_{ti} with the highest BMscore.
- The function of similarity SimCos() is given by equation (25). SimCos(D_{k_i}, D_{k_j}) $\leq$ **β** means that D_{k_i} and D_{k_1} are less similar.

The algorithm is the following (**<u>Algorithm 2</u>**);

1. For each D_{k_i} of C'_{ti} started by D_{k_2}
   a) j = 1

b) While [(SimCos(D_{k_i}, D_{k_j}) $\leq$ **β**) and (j $\leq |C''_{ti}|$)]

- j++

c) If (j > $|C''_{ti}|$))

- $C''_{ti} = C''_{ti} \cup \{D_{k_i}\}$

The result of Algorithm 2 is the subset of C'_{ti} that contains the less similar, relevant and discriminant documents of topic t_i.

$C''_{ti} = \left\{D_{k_1}, \dots, D_{k_l}, \dots, D_{k_\gamma}\right\}$ where $\alpha \geq \gamma$

Step 3: Dynamic updating of model by novelty (addition of new annotated document)

This step verifies whether the new annotated document is relevant to its annotated topics. Remember that $v_{ti}=\{W_1,\dots,W_k,\dots, W_{Ni}\}$ denotes the vocabulary of the topic t_i.

Based on steps 1 and 2, note the vectors IDF^s_{ti}, ITF^s, and TF^s_{ti}:

- $IDF^s_{ti} = (IDF(W_1, C_{ti}), \dots, IDF(W_k, C_{ti}), \dots, IDF(W_{Ni}, C_{ti}))$

where $IDF(W_k, C_{ti})$ denotes the number of documents in the corpus C_{ti} where the term W_k appears at the state s.

- $ITF^s = (ITF(W_1), \dots, ITF(W_k), \dots, ITF(W_{Ni}))$

where $ITF(W_k)$ denotes the number of topics where W_k appears at the state s.

- $TF^s_{ti} = (TF(W_1, t_i), \dots, TF(W_k, t_i), \dots, TF(W_{Ni}, t_i))$

Where $TF(W_k, t_i)$ denotes the number of occurrences of W_k in all the documents of corpus C_{ti} at the state s.

The algorithm for the dynamic updating of the model by novelty **(<u>Algorithm 3</u>)** is defined as follows, where vectors IDF^s_{ti}, ITF^s, and TF^s_{ti} are used as inputs:

1. For a new document *d*,
   a) For each topic t_i of *d*
      - compute the TF-IDF of each term *W* of *d* based on IDF^s_{ti};

$$f(W, d, C_{ti}) = TF - IDF(W, d, C_{ti}) = TF(W, d) * \log\left(\frac{|C_{ti}|}{IDF(W, C_{ti}) + 1}\right)$$

- rank the terms W of d based on their TF-IDF
- select the E_{ti} terms W of d with highest TF-IDF
- compute the TF-ITF of each selected term W of d based on ITF_{ti}^{s} and TF_{ti}^{s}

$$g(W,t_i) = TF - ITF(W,t_i) = [TF(W,t_i) + TF(W,d)] * \log(\frac{|T|}{ITF(W_k)})$$

- classify the term of d by TF-IDF in descending order
- compute the BMscore of d

$$BMscore\ (d) = \sum_{|E_i|} g(W,t_i)$$

- If the BMscore (d) is higher than the smallest BMscore of C'_{ti} document
  - $C'_{\mathrm{ti}} = C'_{ti} \setminus \{D_{k_i}\}$

    where D_{k_i} denotes the document of C'_{ti} with the smallest BMscore
  - $C'_{\mathrm{ti}} = C'_{ti} \cup \{d\}$
  - Call Algorithm 2 to update C''_{ti}

- update vector IDF_{ti}^{s}
  - $IDF(W,C_{ti}) = IDF(W,C_{ti}) + 1$
- update vector TF_{ti}^{s}
  - $TF(W,t_i) = TF(W,t_i) + TF(W,d)$

2. Topic detection phase

- Let G be the BM-SemGraph of the entire collection;
- Let T_d be the list of topics of document d.

The algorithm for the topic detection process phase (**<u>Algorithm 4</u>**) is the following:

1. $T_d = \{\}$
2. For a new document *d*,
   a) Generate BM-SemGraph H of document
   b) For each feature vector of topic t_j^d of BM-SemGraph H
      - Identify the main term W_i using:
        - $BMRank(W_i|t_j^d) = \frac{\sum_{W_k \in A} BMW(W_i, W_k \mid t_j^d)}{|A|}$
      - For each feature vector of topic t_i of BM-SemGraph G

- If W_i is a term of feature vector of topics t_i
  - Compute the similarity between t_i and topic t_j^d as follows:

$$Sim(t_i|t_j^d) = \frac{\sqrt{\sum_{W_k \in B} \left(BMW(W_i, W_k \mid t_i) - BMW\left(W_i, W_k \mid t_j^d\right)\right)^2}}{\sqrt{\sum_{W_h \in C} \left(BMW(W_i, W_h \mid t_i) - BMW\left(W_i, W_h \mid t_j^d\right)\right)^2}}$$

  - If $Sim\left(t_i \middle| t_j^d\right) \leq \text{VectorSimilarityThreshold}$

$$T_d = T_d \cup \left\{(t_i, t_j^d)\right\}$$

3. Topic refining phase

The algorithm for the topic refining process phase (**<u>Algorithm 5</u>**) is the following:

- Let H be the new document *d* term clustering by topic;
- Let G be clusters of terms by topic;
- Let LMatch be the list of clusters of H and G which match ;
- Let LNotMatch be the list of clusters of H and G which do not match.

1. LMatch = {}
2. LNotMatch = {}
3. For each terms cluster C_j^d of topic t_j^d of H
   a) For each term cluster C_i of topic t_i of G
      - NotLinkG = 1 + maximum number of hops between two terms in term cluster C_i of topic t_i of G
      - HopNumberH = 0
      - HopNumberG = 0
      - For each link (W_i; W_j) of terms cluster C_j^d of topic t_j^d in H
        - HopNumberH = HopNumberH + 1
        - Hop = Find the shortest number of hops between W_i and W_j in terms cluster of topic t_i of G

- If Hop = 0

  Hop = NotLinkG

- HopNumberG = HopNumberG + Hop

b) Sim (t_j^d, t_i) = HopNumberH / HopNumberG

c) If Sim (t_j^d, t_i) > Ω

- LMatch = LMatch ∪ $\{(t_i, t_j^d)\}$

Else

- LNotMatch = LM LNotMatch ∪ $\{(t_i, t_j^d)\}$

<u>Algorithm 6</u> is the following:

- Let D_n be the new document;
- Let TS_{Dn} be the list of similar topics associated to D_n;
- Let TD_{Dn} be the list of distinct topics associated to D_n.

1. For a new document D_n
2. For each selected topic t_i of T
   a) $l = 1$
   b) $TD_{Dn} = \{\}$
   c) $TS_{Dn} = \{\}$
   d) While [(SimCos (D_n, D_{k_l}) < **β** and ($l \leq |C''_{ti}|$)] // $D_{k_l} \in C''_{ti}$
      - l++
   e) if ($l \leq |C''_{ti}|$)
      - $TS_{Dn} = TS_{Dn} \cup \{t_i\}$

      Else
      - $TD_{Dn} = TD_{Dn} \cup \{t_i\}$

Appendix B: BM-SSEA Processes, Phases and Algorithms

1. BMEmoSenMod generation phase

This step makes use of the corpus of documents rated over E emotions. However, it is feasible to perform this step periodically in order to update the sentiment and emotion lexicon (e.g., BMEmoSenMod).

<u>Algorithm 7</u>

Input: WordNet, WordNet-Affect, SentiWordNet and NRC emotion lexicon

Output: BMEmoSenMod

Emotions	**Topic**	**Word feature**	**Emotion probability of f_j**	**Sentiment probability of f_j**
…		…		
e_i		1		
		…		
	t_k	f_j	$\text{EmoPro}(e_i \mid f_j, t_k)$	$\text{SenPro}(s_i \mid f_j, t_k)$
		…		
…		…		
E_E		…		

1. For each emotion $\boldsymbol{e_i}$ BMEmoSenMod
   a. Identify the sample contents related to emotion $\boldsymbol{e_i}$
   b. Extract the keywords W_j from the documents $\{C_1, \ldots, C_h, \ldots, C_q\}$
   c. Associate with each word-POS a feeling score to the keyword W_j to obtain the word feature f_j
   d. Detect the topic t_k of document d where W_j appears
   e. Compute the emotion probability of the obtained word feature f_j of keyword W_j

$$\text{EmoPro}(e_i \mid f_j, t_k) = Val(f_j) \times \frac{\sum_{d \in C_{tk} \subset ND} p(f_j, t_k, d) \times oc(e_i, t_k)}{\sum_{e_l \in E} \sum_{d \in C_{tk} \subset ND} p(f_j, t_k, d) \times oc(e_l, t_k)}$$

f. Compute the sentiment probability of the obtained word feature f_j of keyword W_j

$$\mathrm{SenPro}(s_i|f_j,t_k) = SSco(f_j) \times \frac{\sum_{\mathrm{d} \in C_{tk} \subset \mathrm{ND}} p(f_j,t_k,d) \times oc(s_i,t_k)}{\sum_{s_l \in S} \sum_{\mathrm{d} \in C_{tk} \subset \mathrm{ND}} p(f_j,t_k,d) \times oc(s_l,t_k)}$$

g. Add $\mathrm{EmoPro}(e_i|f_j,t_k)$ and $\mathrm{SenPro}(s_i|f_j,t_k)$ in the sentiment and emotion lexicon BMEmoSenMod

2. Sentiment and emotion discovery

This step is performed for a new document targeted to discover its sentiments and emotions.

<u>Algorithm 8</u>

Input: new document and BMEmoSenMod

Output: emotional vector of new document

- Let D be the given document
- Extract the word feature f_j of D

1. For each word feature f_j of D
   a. If f_j is in the sentiment and emotion lexicon BMEmoSenMod,
      - For each associated emotion $\boldsymbol{e_i}$
        $\mathrm{W_E}(\mathrm{ND}, e_i) = \sum_{\mathrm{W_j} \in \mathrm{W_{ND}}} \mathrm{E}(f_j, \mathrm{ND}, e_i)$
   b. Else
      - Identify the synonyms f_y of f_j in the BMEmoSenMod
      - For each associated emotion $\boldsymbol{e_i}$
        $\mathrm{W_E}(\mathrm{ND}, e_i) = \frac{\sum_{\mathrm{W_y} \in \mathrm{BMEmoSenMod}} \mathrm{E}(f_y, \mathrm{ND}, e_i)}{m}$
        // m denotes the number of synonyms of f_j

2. Normalization of each $\mathrm{W_E}(\mathrm{ND}, e_i)$
3. Return ($\mathrm{W_E}(\mathrm{ND}, e_1)$, …,$\mathrm{W_E}(\mathrm{ND}, e_i)$, …,$\mathrm{W_E}(\mathrm{ND}, e_E)$,$\mathrm{W_E}(\mathrm{ND}, s_P)$,$\mathrm{W_E}(\mathrm{ND}, s_N)$)

Appendix C: Semantic topic detection

Semantic topic detection, a fundamental aspect of SIR, helps users to efficiently detect meaningful topics. It has attracted significant research in several communities in the last decade, including public opinion monitoring, decision support, emergency management and social media modeling (Hurtado et al., 2016; Sayyadi & Raschid, 2013). STD is based on large and noisy data collections such as social media, and addresses both scalability and accuracy challenges. Initial methods for STD relied on clustering documents based on a core group of keywords representing a specific topic, where, based on a ratio such as tf-idf, documents that contain these keywords are similar to each other (Niu et al., 2016; Salton & Buckley, 1988). Next, variations of tf-idf were used to compute keyword-based feature values, and cosine similarity was used as a similarity (or distance) measure to cluster documents. The following generation of STD approaches, including those based on latent Dirichlet allocation (LDA), shifted analysis from directly clustering documents to clustering keywords. Some examples of these advances in STD are presented in (David M. Blei et al., 2003).

However, social media collections differ along several criteria, including the size distribution of documents and the distribution of words. One challenge is to rapidly filter noisy and irrelevant documents, while at the same time accurately clustering a large collection. Bijalwan et al. (Bijalwan et al., 2014), for example, experimented with machine learning approaches for text and document mining and concluded that k-nearest neighbors (KNN), for their data sets, showed the maximum accuracy as compared to naive Bayes and term-graph. The drawback for KNN is that time complexity (i.e., amount of time taken to run) is high but it demonstrates better accuracy than others.

In the last decade, semantic topic detection has attracted significant research in several communities, including information retrieval. Generally, a topic is represented as a set of descriptive and collocated keywords/terms. Initially, document clustering techniques were adopted to cluster content-similar documents and extract keywords from clustered document sets as the representation of topics (subjects). The predominant method for topic detection is the latent Dirichlet allocation (LDA) (David M. Blei et al., 2003), which assumes a generating

process for the documents. LDA has been proven a powerful algorithm because of its ability to mine semantic information from text data. Terms having semantic relations with each other are collected as a topic. LDA is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities. In the context of text modeling, topic probabilities provide an explicit representation of a document.

The literature presents two groups of text-based topic detection approaches based on the size of the text: short text (Cigarrán et al., 2016; Cotelo et al., 2016; Dang et al., 2016; Hashimoto et al., 2015) such as tweets or Facebook posts, and long text (David M. Blei et al., 2003; Bougiatiotis & Giannakopoulos, 2016; P. Chen et al., 2016; Salatino & Motta, 2016; Sayyadi & Raschid, 2013; C. Zhang et al., 2016) such as a book.

For example, Dang et al. (Dang et al., 2016) proposed an early detection method for emerging topics based on dynamic Bayesian networks in micro-blogging networks. They analyzed the topic diffusion process and identified two main characteristics of emerging topics, namely attractiveness and key-node. Next, based on this identification, they selected features from the topology properties of topic diffusion, and built a DBN-based model using the conditional dependencies between features to identify the emerging keywords. But to do so, they had to create a term list of emerging keyword candidates by term frequency in a given time interval.

Cigarran et al. (Cigarrán et al., 2016) proposed an approach based on formal concept analysis (FCA). Formal concepts are conceptual representations based on the relationships between tweet terms and the tweets that have given rise to them.

Cotelo et al. (Cotelo et al., 2016), when addressing the tweet categorization task, explored the idea of integrating two fundamental aspects of a tweet: the textual content itself, and its underlying structural information. This work focuses on long text topic detection.

Recently, considerable research has gone into developing topic detection approaches using a number of information extraction techniques (IET), such as lexicon, sliding window, boundary techniques, etc. Many of these techniques (P. Chen et al., 2016; Salatino & Motta, 2016;

Sayyadi & Raschid, 2013; C. Zhang et al., 2016) rely heavily on simple keyword extraction from text.

For example, Sayyadi and Raschid (Sayyadi & Raschid, 2013) proposed an approach for topic detection, based on keyword-based methods, called KeyGraph, that was inspired by the keyword co-occurrence graph and efficient graph analysis methods. The main steps in the KeyGraph approach are as follows:

1. The first step is construction of a keyword co-occurrence graph, called a KeyGraph, which has one node for each keyword in the corpus and where edges represent the co-occurrence of the corresponding keywords weighted by the count of the co-occurrences;
2. Secondly, making use of an off-the-shelf community detection algorithm, community detection is taken into account where each community forms a cluster of keywords that represent a topic. The weight of each keyword in the topic feature vector is computed using the tf-idf formula. The TF value is computed as the average co-occurrence of each keyword from the community with respect to the other keywords in that community;
3. Then, to assign a topic to a document, the likelihood of each topic t with the vector of keyword ft is computed using the cosine similarity of the document;
4. Finally, for each pair of topics, where multiple documents are assigned to both topics, it is assumed that these are subtopics of the same parent topic and are therefore merged.

In other words, KeyGraph is based on the similarity of keyword extraction from text. We note two limitations to the approach, which requires improvement in two respects. Firstly, they failed to leverage the semantic information derived from topic model. Secondly, they measured co-occurrence relations from an isolated term-term perspective; that is, the measurement was limited to the term itself and the information context was overlooked, which can make it impossible to measure latent co-occurrence relations.

Salatino and Motta (Salatino & Motta, 2016) suggested that it is possible to forecast the emergence of novel research topics even at an early stage and demonstrated that such an

emergence can be anticipated by analyzing the dynamics of pre-existing topics. They presented a method that integrates statistics and semantics for assessing the dynamics of a topic graph:

1. First, they select and extract portions of the collaboration networks related to topics in the two groups a few years prior to the year of analysis. Based on these topics, they build a topics graph where nodes are the keywords while edges are the links representing co-occurrences between keywords;
2. Next, they transform the graphs into sets of 3-cliques. For each node of a 3-clique, they compute the weight associated with each link between pairs of topics by using the harmonic mean of the conditional probabilities. While this is a satisfactory approach to find latent co-occurrence relations, the approach assumes that keywords are topics.

Chen et al. (P. Chen et al., 2016) proposed a novel method for hierarchical topic detection where topics are obtained by clustering documents in multiple ways. They used a class of graphical models called hierarchical latent tree models (HLTMs). Latent tree models (LTMs) are tree-structured probabilistic graphical models where the variables at leaf nodes are observed and the variables at internal nodes are latent. It is a Markov random field over an undirected tree carried out as follows:

1. First, the word variables are partitioned into clusters such that the words in each cluster tend to co-occur and the co-occurrences can be properly modeled using a single latent variable. The authors achieved this partition using the BUILDISLANDS subroutine, which is based on a statistical test called the uni-dimensionality test (UD-test);
2. After the islands are created, they are linked up so as to obtain a model over all the word variables. This is carried out by the BRIDGEISLANDS subroutine, which estimates the mutual information between each pair of latent variables in the islands. This allows construction of a complete undirected graph with the mutual information values as edge weights, and finally the maximum spanning tree of the graph is determined (P. Chen et al., 2016).

Hurtado et al. (Hurtado et al., 2016) proposed an approach that uses sentence-level association rule mining to discover topics from documents. Their method considers each sentence as a transaction and keywords within the sentence as items in the transaction. By exploring

keywords (frequently co-occurring) as patterns, their method preserves contextual information in the topic mining process. For example, whenever the terms: “machine”, “support” and “vector” are discovered as strongly correlated keywords, either as “support vector machine” or “support vector”, they assumed that these patterns were related to one topic, i.e., “SVM”. In order to discover a set of strongly correlated topics, they used the CPM-based community detection algorithm to find groups of topics with strong correlations. As in (P. Chen et al., 2016), their contribution was limited to simulating existing algorithms.

Zhang et al. (C. Zhang et al., 2016) proposed LDA-IG, an extension of KeyGraph (Sayyadi & Raschid, 2013). It is a hybrid relations analysis approach integrating semantic relations and co-occurrence relations for topic detection. Specifically, their approach fuses multiple types of relations into a uniform term graph by incorporating idea discovery theory with a topic modeling method.

1. Firstly, they defined an idea discovery algorithm called IdeaGraph that was adopted to mine latent co-occurrence relations in order to convert the corpus into a term graph.
2. Next, they proposed a semantic relation extraction approach based on LDA that enriches the graph with semantic information.
3. Lastly, they make use of a graph analytical method to exploit the graph for detecting topics. Their approach has four steps:
    a. Pre-processing to filter noise and adjust the data format suitable for the subsequent components;
    b. Term graph generation to convert the basket dataset into a term graph by extracting co-occurrence relations between terms using the Idea Discovery algorithm;
    c. Term graph refining with semantic information using LDA to build semantic topics and tp-izp, inspired by tf-idf, to measure the semantic value of any term in each topic;
    d. Topic extraction from the refined term graph by assuming that a topic is a filled polygon and measuring the likelihood of a document d being assigned to a topic using tf-idf. However, their approach does not include machine learning, which would allow the framework to find new topics itself.

From our review of related work, we conclude that the main drawbacks of existing approaches to topic detection are as follows:

1. They are based on simple keyword extraction from text and lack semantic information that is important for understanding the document. To tackle this limitation, our work uses semantic annotations to improve document comprehension time;
2. Co-occurrence relations across the document are commonly neglected, which leads to incomplete detection of information. Current topic modeling methods do not explicitly consider word co-occurrences. Extending topic modeling to include co-occurrence can be a computational challenge. The graph analytical approach to this extension was only an approximation that merely took into account co-occurrence information alone while ignoring semantic information. How to combine semantic relations and co-occurrence relations to complement each other remains a challenge;
3. Existing approaches focus on detecting prominent or distinct topics based on explicit semantic relations or frequent co-occurrence relations; as a result, they ignore latent co-occurrence relations. In other words, latent co-occurrence relations between two terms cannot be measured from an isolated term-term perspective. The context of the term needs to be taken into account;
4. More importantly, even though existing approaches take into account semantic relations, they do not include machine learning to find new topics automatically.

The main conclusion is that most of the existing related research is limited to simulations using existing algorithms. None contribute improvements to detect topics more accurately.

APPENDIX III

An Assisted Literature Review using Machine Learning Models to Build a Literature Corpus and Recommend References Based on Corpus Radius

Ronald Brisebois[1], Alain Abran[2], Apollinaire Nadembega[1], Philippe N'techobo[1]

[1] Bibliomondo, Montréal, Canada
{ronald.brisebois,apollinaire.nadembega,philippe.ntechobo}@bibliomondo.com
[2] École de technologie supérieure, Université du Québec, Canada,
alain.abran@etsmtl.ca

Paper submitted for publication to Information Retrieval Journal, January 2017

Abstract

With the evolving of research and huge volume papers, there is a need to assist researchers in the manual process of building literature review (LR). This paper proposes an assisted literature review (ALR) prototype (STELLAR - Semantic Topics Ecosystem Learning-based Literature Assistant Review). Using text and data mining models (TDM), machine learning models (MLM) and classification model, all of which learn from researchers' annotated data and semantic enriched metadata (SMESE), STELLAR helps researchers discover, identify, rank and recommend relevant papers for an ALR according to the researcher selection. Considering more criteria (venue age and impact, citation category and polarity, researchers' annotated data, authors' impact and affiliation institute, etc.) than existing approaches, STELLAR evaluates papers and related bibliographic attributes in order to determine their relevancy and aggregates all relevant components into an assisted literature review object (ALRO).

This paper presents the MLM and algorithms that:

- Identify relevant papers based on key finding, citation and paper feature impact.

- Compute papers semantic similarity with the researcher selection parameters.
- Assist the researcher in refining and recommending the list of papers relevant.
- Aggregate all relevant components into an ALRO.

STELLAR performance was compared to existing approaches using a number of simulations.

Keywords: assisted literature review, literature review, machine learning, literature review enrichment, semantic topic detection, text and data mining.

1. Introduction

Electronic access to research papers plays a primordial role in the dissemination of research results published in conference proceedings, journals and new platforms such as researcher media. Literature reviews, in which publications are selected by relevancy and evaluated, are a fundamental component of scientific writing. But the huge volume of scientific publications available is becoming an issue for researchers (Boote & Beile, 2005; Mayr et al., 2014): given that their time is limited, it is becoming impossible for researchers to read and carefully evaluate every publication within their own specialized field.

A manual literature review (LR) process is very labor intensive, and the time that researchers must dedicate to searching for literature will vary according to their research topic. For instance, Gall et al. (Gall et al., 1996) estimate that a decent LR for a dissertation takes three to six months to complete. In their academic process, postgraduate students in all disciplines need to be able to write an accurate LR. Whether a short review as an assignment in a Master's program, or a full-length LR for a PhD thesis, students find it difficult to produce a LR with all of the relevant and up-to-date papers. Researchers also have to stay aware of newly published papers on related topics to produce a meaningful LR.

An LR is not simply a summary of what is published about a particular topic; it must address a research question and must identify primary sources and references. It should focus only on the relevant literature available from all literature, that is, on references collected from recognized experts on the topic or related topics. According to (Carlos & Thiago, 2015; Gulo et al., 2015), an LR process consists in locating, appraising and synthesizing the best available empirical evidence to answer specific research questions. An LR will look at as much existing research as is feasible and will review scholarly papers and theses in the relevant area. It is a state-of-the-art search and evaluation of the available literature on a given topic or concept. It is not a chronological description of what has been discovered; it has to provide an analytical overview of the significant and relevant literature published on the topic. An ideal LR should retrieve all relevant papers for inclusion and exclude all irrelevant papers (Carlos & Thiago, 2015; Gulo et al., 2015).

The researcher's main tasks in producing a manual LR are as follows:

1. Clearly identify the topic or field of research;
2. Search, survey and evaluate the available literature;
3. Identify and understand the keywords, vocabulary, definitions, concepts and terms using an appropriate specialized dictionary, i.e., one that pertains to the topic or field in question;
4. Order the relevant works within the context of their contribution to the LR;
5. Present the literature in an organized way;
6. Identify the main methodologies and research techniques used in the works;
7. Summarize, synthesize and integrate the relevant works by abstracting their content.

The sources and references have to be relevant, as current as possible and cited in a format appropriate to the discipline and publication sources.

The aim of the paper presented here is to help the researcher identify references relevant a Literature Corpus for the LR, that is, the first four of the seven tasks listed above. The remaining three tasks will be addressed in a future paper.

The following questions are essential to building a good LR:

1. What are the origins, definitions and detailed description of the topic or concept?
2. For each paper, what are the author's credentials and relevancy in regard to the topic discussed (e.g., number of papers and citations related to the topic)?
3. What are the proceedings or journal's credentials and its relevancy to the topic?
4. What is the reputation or ranking of the publisher?
5. When the LR is spread over a number of years, it is important to decide which references to include. This means determining how many years from the current date the content will be retained in the analysis.
6. If the researcher's project is multi-year, how to ensure that the LR stays up to date for a specific topic over the duration of the project?
7. What are the main conclusions from previous works on this topic?

To manually find sources of content for the LR, the first step is to identify the relevant topics or concepts and prioritize them. A way to identify the relevant ones is to check the lists of references to see which are frequently cited and how often. This requires ranking the LR references according to the specific research topic or concept and other parameters such as publication date, sources, etc.

With the massive increase in digital content and widespread use of search engines, the number of returned results can be tremendous—which then makes it challenging to select only the papers relevant to the LR topic. This has led to the emergence of result ranking algorithms defined as the procedure used by search engines to assign priorities to returned results.

In the context of scientific content, the ranking algorithms for content evaluation are referred to as scientometrics or bibliometrics (Beel et al., 2013; Bornmann et al., 2014, 2015; Cataldi et al., 2016; Dong et al., 2016; Franceschini et al., 2015; Hasson et al., 2014; Madani & Weber, 2016; Marx & Bornmann, 2016; MASIC & BEGIC, 2016; Packalen & Bhattacharya, 2015; Rúbio & Gulo, 2016; Wan & Liu, 2014; S. Wang et al., 2014; M. Zhang et al., 2015).

With the interdisciplinary nature of research and electronic access to papers, there is a need to facilitate and assist researchers in the iterative creation of their LRs. Semantic metadata allow more accurate searching than keywords and may help to get better relevant results for an

assisted literature review (ALR). Semantic metadata can be extracted using text and data mining (TDM) algorithms. TDM, machine learning models (MLM) have been designed to learn from papers and researchers' annotated papers and to identify relevant papers for a specific topic and research field.

In this paper, we report on our work to define and build an assisted LR prototype designed to reduce reading load by pointing the researcher to a recommended selection of documents. This paper proposes an ALR prototype (referred to here as STELLAR), i.e., a set of TDM and MLM for searching, discovering, ranking and recommending papers for an ALR. For instance, STELLAR will assess citations and other bibliographic attributes in order to select and rank papers and include them (or not) in the list of recommended references for the researcher.

A prototype of STELLAR has been implemented using a software ecosystem described in SMESE V1 (Brisebois, Abran, & Nadembega, Unpublished results) and SMESE V3 (Brisebois, Abran, Nadembega, & N'techobo, Unpublished results). The remainder of the paper is organized as follows.

1. Section 2 presents the related works;
2. Section 3 describes the STELLAR multi-platform architectural model included in the SMESE prototype;
3. Section 4 presents the MLM designed for the STELLAR prototype;
4. Section 5 presents an evaluation of the prototype through a number of ALR simulations;
5. Section 6 contains a summary and suggestions for future work.

2. Related Works

This section presents the related works in the following sequence:

1. Ranking of scientific papers,
2. Text and data mining, and more specifically:
    a. Machine learning models (MLM),
    b. Automatic text summarization (ATS),
    c. Automatic multi-documents summarization for ALRs.

3. Assisted literature review object (ALRO).

2.1 Ranking of scientific papers

The proliferation of scientific publications and the online availability of repositories make it challenging for researchers to produce and maintain an updated bibliography for specific research fields. Within this context, there is an increasing need to develop software tools that can facilitate and aid LR automation and optimization. Unfortunately, few works have explored how to assist researchers in building a LR.

Two means of quantitatively evaluating scientific research output are discussed in the literature: peer-review and citation-based bibliometrics indicators. The main limitation of peer-review-based approaches is the subjectivity of evaluators, while citations-based approaches have been criticized for having a scope limited to academia and neglecting the broader societal impact of research (Marx & Bornmann, 2016).

According to the literature, citation analysis is widely used to measure scientific papers and their impact. Recently some iterative processes, such as PageRank, have been applied to citation networks to perform this function. Unfortunately, the PageRank algorithm also has some limitations: for example, recent papers not yet cited do not appear in the top level of results. Furthermore, the links between papers are oriented to a single direction: from a citing paper to cited papers.

Scientific paper ranking should also depend on the venue, the location of publication, the year, the author and the citation index. Some works in the field of scientific impact evaluation (Bornmann et al., 2014, 2015; Cataldi et al., 2016; M. Zhang et al., 2015) address the ranking of universities, institutions and research teams. For instance, M. Zhang et al. (M. Zhang et al., 2015) propose a method to discover and rank collaborative research teams based on social network analysis in combination with traditional citation analysis and bibliometrics. In this approach, the research teams are ranked using indexes including both scientific research outcomes and the close degree of co-author networks.

For this research, many existing approaches for scientific paper ranking have been evaluated (Bornmann et al., 2014, 2015; Gulo et al., 2015; Hasson et al., 2014; Madani & Weber, 2016; Marx & Bornmann, 2016; Rúbio & Gulo, 2016; Wan & Liu, 2014; S. Wang et al., 2014). They suffer from a number of limitations:

1. Most existing approaches focus on the researcher index or journal index to evaluate scientific research impact, ignoring the papers index—the most important metric for measuring the impact of a paper;
2. Of the approaches that do focus on the papers index, most only use the citations count; in addition, they do not consider the age of papers, penalizing the recent ones;
3. The few approaches focusing on the evaluation of papers themselves do not take into account the Social Level Metric, and they do not consider the category or polarity of citations;
4. Some approaches make use of journal information to rank papers; however, they do not consider the other types of venues, such as conference proceedings, workshops or unpublished documents such as technical reports;
5. Several approaches make use of MLM; however, they require a large manual contribution from specialists or experts to train the learning model;
6. Very few works focus on text-based analysis to identify relevant papers, and those that do are limited to titles and abstracts.

A comparison of two approaches proposed in the literature for scientific paper ranking is presented in Table A 3.1: PTRA (Hasson et al., 2014) and ID3 (Rúbio & Gulo, 2016):

1. PTRA: Hasson et al. (Hasson et al., 2014) propose a ranking algorithm, called Paper Time Ranking Algorithm (PTRA), that depends on three factors: paper age, citation index and publication venue with a different priority assigned to each one of them. For a given paper, they compute its weight as the sum of the age of the conference proceedings or the journal impact factors, the number of citations of the paper and the age of paper;
2. ID3: Rúbio and Gulo (Rúbio & Gulo, 2016) propose recommending papers based on known classification models, including the paper's content and bibliometric features. Indeed, they combine text mining, ML algorithms and bibliometric measures to

automatically classify the relevant papers. They make use of the paper's metadata (such as year of publication, citation number, reference number and publication venue) to measure the paper's relevancy to specific field. To apply the ML algorithm, they make use of specialist annotations.

It can be seen from Table A 3.1 that in ranking and identifying relevant contributions, neither of these two approaches takes into account author impact, citation category, venue impact, authors' institutes or citing documents (the six rightmost columns).

Table A 3.1 The PTRA and ID3 approaches for ranking papers

Approaches	Year of publication	Citation number	Reference	Venue type	Venue age	Authors' impact	Citation category	Venue impact	Authors' institutes	Citing document of cited document
PTRA (Hasson et al., 2014)	X	X		X						
ID3 (Rúbio & Gulo, 2016)	X	X	X	X						

2.2 Text and data mining (TDM)

In scientific research, documents (such as journal papers, conference proceedings or research reports) have a specific organization and relevant sections that are different from other types of documents such as narrative text (R. Zhang et al., 2016).

The purpose of a text summarizer is to select the most important facts and present them in a sensible order while avoiding repetition (Carenini et al., 2013). However, scientific papers frequently contain repeated expressions and sentences. Consequently, narrative text summarization approaches are not adequate for summarizing scientific papers for an ALR;

however, the principles of automatic text summarization (ATS) may be extended to apply here. This sub-section therefore reports on work dealing with:

1. MLM,
2. ATS,
3. Automatic multi-documents summarization for LR.

2.2.1 Machine learning models (MLM)

MLM is a subfield of computer science that evolved from the study of pattern recognition and computational learning theory in artificial intelligence. MLM explores the definition and study of algorithms that can learn from and make predictions on data. Tom Mitchell, in his book Machine Learning (Mitchell, 1997), provides a definition in the opening line of the preface: "The field of machine learning is concerned with the question of how to construct computer programs that automatically improve with experience."

There are three different axes for MLM:

1. Text and data mining: using historical data to improve decisions:
   a. Medical records → medical knowledge,
   b. Document notices → document knowledge.

2. Software algorithms that are difficult to program by hand:
   a. Image recognition and classification,
   b. Filtering algorithms/news feeds,
   c. Sort the answers according to their relevancy to a dynamic query,
   d. Optical character recognition,
   e. Bibliographic classification.

3. User modeling:
   a. Automatic recommender assistants,
   b. Personal assistants such as Google Now and Apple Siri.

In the context of TDM, MLM is used mainly for metadata enrichment and literature review refinement in the context of ALR. Indeed, for literature summarization, two main MLM trends are identified:

1. Supervised systems that rely on ML algorithms trained on pre-existing document-summary pairs, namely:
   a. Linear algorithms for classification and regression,
   b. Non-linear algorithms for decision tree, rule-based and neural networks.

2. Unsupervised techniques based on properties and heuristics derived from the text. The unsupervised summarization methods (Z. He et al., 2015) are mainly based on the weight of words in sentences, as well as the sentence position in a document.

For example, Carlos and Thiago (Carlos & Thiago, 2015) developed a supervised MLM-based solution for text mining scientific articles using the R language in "Knowledge Extraction and Machine Learning" based on social network analysis, topic models and bipartite graph approaches. Indeed, they defined a bipartite graph between documents and topics that makes use of the Latent Dirichlet Allocation topic model.

In regards to the classification and ranking problem, there are different MLM. To determine which model performs best, the best way remains the use of prototypes.

An MLM can also be dynamic, meaning that it can train itself on the analysis of new data. In the case of MLM's K-means clustering algorithm, the data would be classified into clusters and any new metadata and data would clarify the cluster boundaries, thus improving the model's ability to classify accurately.

The next two sub-sections report on MLM for single or multi-document text summarization.

2.2.2 Automatic text summarization (ATS)

Document key phrases enable fast and accurate searching for a given document within a large collection, and have exhibited their potential for improving many natural language processing

and semantic information retrieval tasks, such as automatic text summarization (ATS) and ALR. ATS has received a lot more attention than ALR.

According to (Saggion & Poibeau, 2013), there are two main types of ATS:

1. Extractive summarization selects the important sentences from the original input documents to form a summary;
2. Abstractive summarization (Genest & Lapalme, 2012; Gerani et al., 2014) paraphrases the corpus using novel sentences that usually involve information fusion, sentence compression and reformulation. Although an abstractive summary could be more concise, it requires deep natural language processing techniques.

According to (Ferreira et al., 2013), sentence scoring is the technique most used for extractive text summarization. In general, there are three possible approaches:

1. Word scoring, which assigns scores to the most important words;
2. Sentence scoring, which examines the features of a sentence such as its position in the document, similarity to the title, etc;
3. Graph scoring, which analyzes the relationships between sentences.

Extractive summaries are therefore more feasible and practical, and so this sub-section focuses on that type of ATS. (Nenkova & McKeown, 2012) identified three relatively independent tasks performed by almost all extractive summarizers:

1. Create an intermediate representation of the input which captures only the key aspects of the text;
2. Score sentences based on that representation ;
3. Select a summary consisting of several sentences.

For the intermediate representation task, they identified the following approaches:

1. Topic representation approaches convert the text to an intermediate representation capturing the topics discussed. Such approaches are based on term frequency–inverse document frequency (TF-IDF), topic words, lexical chains, latent semantic analysis, and Bayesian topic models. Each sentence receives a score determined by the extent to which it expresses key topics in the document;

2. Indicator representation approaches represent each sentence in the input according to a list of indicators of importance such as sentence length, location in the document, presence of certain phrases, etc. The sentence score is determined by combining the evidence from the different indicators;
3. Graph models approaches such as LexRank represent the entire document as a network of inter-related sentences. In LexRank, the weight of each sentence is derived by applying stochastic techniques to the graph representation of the text. Finally, the summary is produced through the selection of important sentences.

For the selection of sentences that may be candidates for summarization, the authors refer to three approaches:

1. Best n,
2. Maximal marginal relevancy,
3. Global selection.

In the literature, various solutions for ATS are proposed (CELEBI & DOKUN, 2015; Fang et al., 2015; Hasan & Ng, 2014; Z. He et al., 2015; Ledeneva et al., 2014; Mendoza et al., 2014; Premjith et al., 2015; Sankarasubramaniam et al., 2014); however, several drawbacks can be noticed:

1. Some contributions are greedy in terms of processing time, due to their optimization processes;
2. Some of them make assumptions, such as availability of document topic factors, to validate their approaches;
3. Basic ATS approaches cannot be applied to scientific papers; they need to be adapted to take into account the specificities of scientific papers in terms of document organization and frequently recurring expressions.

2.2.3 Automatic multi-document summarization for ALR

Several approaches have been proposed for scientific paper summarization (Caragea et al., 2014; Carlos & Thiago, 2015; J. Chen & Zhuge, 2014; Conroy & Davis, 2015; Dunne et al.,

2012; Dyas-Correia & Alexopoulos, 2014; Huang & Wan, 2013; Mohammad et al., 2009; Pedram & Omid, 2015; Ronzano & Saggion, 2016; Widyantoro & Amin, 2014). For an ALR, numerous publications need to be analyzed and summarized: this is referred to as multi-document summarization. In the context of scientific research, given a set of scientific papers, multi-document summarization can be used to generate an ALR; however, there are different styles of LR. According to (Jaidka et al., 2010), there are two main styles:

1. A descriptive LR presents a critical summary of a research domain: it summarizes individual papers/studies and provides more information about each one, such as its research methods and results. The descriptive LR focuses on previous studies in terms of approach, results and evaluation, and uses sentence templates to perform rhetorical functions;
2. An integrative LR focuses on the ideas and results extracted from a number of research papers and provides fewer details about individual papers/studies.

For researchers with less experience, a descriptive LR with more details about individual studies is more relevant. For those who prefer to understand the bigger picture and the main research themes, an integrative LR is more relevant. In this contribution, the focus is on recommending a list of relevant, descriptive and enriched papers to help researchers to build their ALRs.

2.3 Assisted literature review object (ALRO)

We have coined the term "assisted literature review object" (ALRO) to refer to a component type that includes many types of metadata and content related to the researchers' specific requests; for example, an ALRO may enrich an ALR with a video or speech that facilitates understanding of the topic of a paper. Indeed, an ALRO is built for a given research topic and differs according to the selection parameters, paper annotations and the time of the request. In other words, it is dynamic, and it aggregates data and enriches metadata about a given ALR to help researchers learn about their field more quickly. Very few works have examined ALRO as defined in this way. In one of these works, Dunn et al. (Dunne et al., 2012) present the results of their effort to integrate statistics, text analytics and visualization in a prototype

interface for researchers and analysts. Their prototype system, called Action Science Explorer (ASE), provides an environment for demonstrating principles of coordination and conducting iterative usability tests with interested and knowledgeable researchers. According to these authors, ASE is designed to support exploration of a collection of papers by rapidly providing a summary, while identifying key papers, topics and research groups. The first drawback of ASE is that it does not propose an algorithm or model for evaluating a scientific paper's relevancy to its research field, but uses only the paper's bibliometric ranking. Also, the authors do not explain how ASE extracts the sentences containing the citations and their locations from the full text of each paper.

From the review of related works, the main drawbacks of existing approaches to ALR are as follows:

1. Regular text summarization techniques cannot be applied to scientific research papers; indeed, such papers have a specific structural organization different from that of other types of documents such as narrative or biographical texts. Conventional TS approaches must therefore be adapted to take into account the specificities of scientific papers in terms of document organization and rhetorical devices;
2. Most of the existing approaches focus only on single paper summarization;
3. Existing works ignore the identification of scientific papers related to the researcher's selection and annotation in terms of research domain, specific topic, matching keywords and subject of research;
4. Finally, existing contributions do not propose an ALRO.

In this research work, we address several limitations of existing approaches (Agarwal et al., 2011; J. Chen & Zhuge, 2014; Dunne et al., 2012; Jaidka et al., 2010, 2013a, 2013b; Patil & Mahajan, 2012; Yeloglu et al., 2011; Zajic et al., 2007) for the design of a better ALR for researchers, including:

1. Ranking of scientific papers,
2. Reviewing of the recommended references for an ALR.

3. STELLAR Multi-platform Architectural Model

This section first presents an overview of the STELLAR (Semantic Topics Ecosystem Learning-based Literature Assisted Review) multi-platform architectural model and a prototype of this architectural model based on SMESE (Semantic Metadata Enrichments Software Ecosystem). The various MLM designed for STELLAR will then be described, including:

1. Discovery ALR,
2. Search & Refine ALR,
3. Assist & Recommend ALR.

3.1 Workflows of manual and assisted literature reviews

The workflow of a manual LR is presented in Figure A 3.1 and the architectural model for an ALR is presented in Figure A 3.2. Within these figures, the white boxes represent manual activities while the shaded ones represent automated activities.

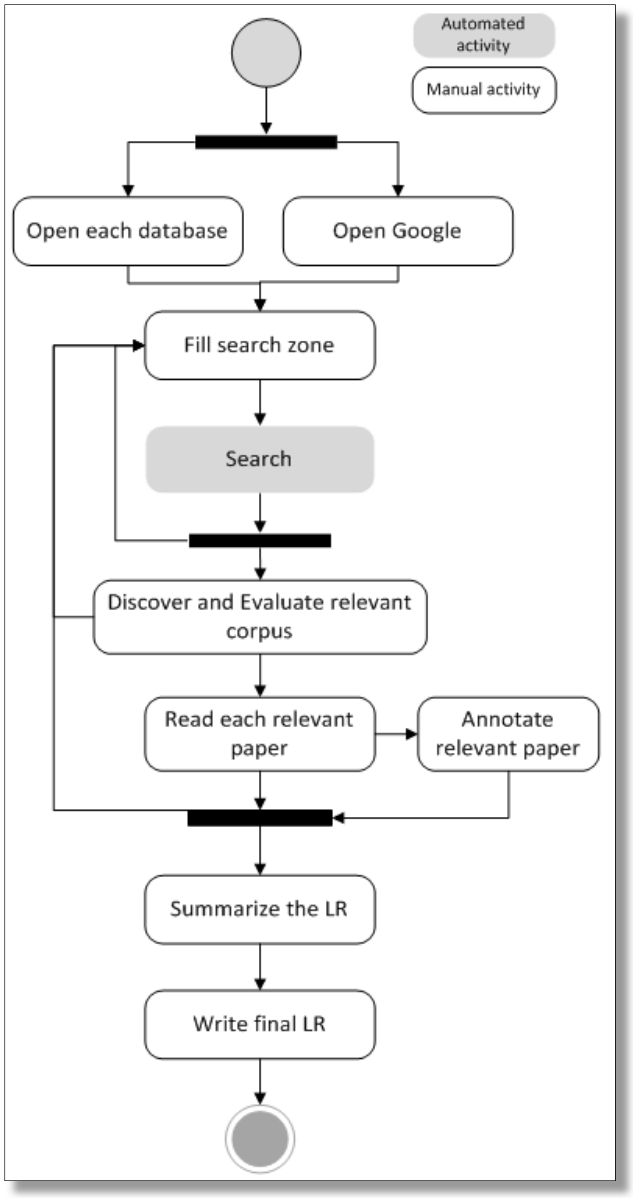


Figure A 3.1 Workflow of a manual LR

An assisted LR (ALR), as illustrated in Figure A 3.2, should allow the following functions:

1. Searching and refining an ALR,
2. Evaluating an LR,
3. Discovering an ALR,
4. Searching in an universal repository, which we will call the universal research document repository (URDR),

5. Searching within an existing ALR, which we will refer to as an ALRO, which is basically a component type with many types of information related to the ALR.

In addition, it should alert the researchers about new papers of interest, related publications or new papers relevant to their ALR.

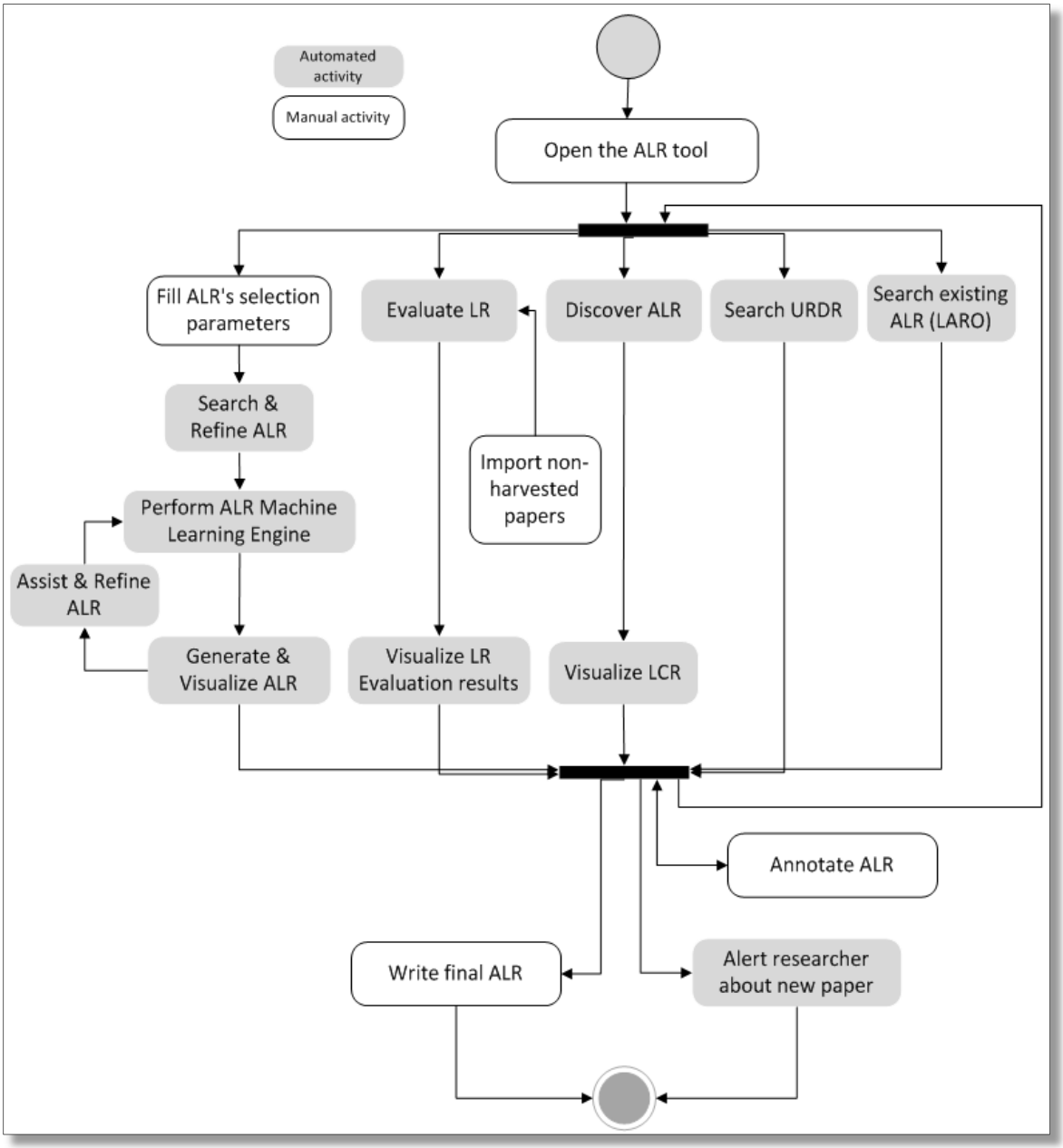


Figure A 3.2 Workflow of an assisted LR (ALR)

In the rest of this section, the STELLAR multi-platform prototype of an ALR is described in more detail.

3.2 Overview of the STELLAR prototype of an assisted LR (ALR)

A literature search has to be systematic and evaluative: it should assess each paper to determine its ranking and whether or not it is worth including in the LR. One of the aims of an ALR is to reduce the reading load by enabling the researcher to read and exploit only a relevant selection of papers.

The models and algorithms of the proposed prototype consist of:

1. TDM models,
2. MLM,
3. A classification model.

This STELLAR prototype (see Figure A 3.3) uses as inputs:

1. A universal research document repository (URDR),
2. The papers annotated by the researcher and previous researchers.

It learns from researchers' annotated papers and the URDR to recommend relevant papers for a specific research field and topic in order to facilitate the creation of a new ALR.

The four main parts of version 1 (V1) of the proposed STELLAR prototype are presented in Figure A 3.3 and explained in the following four sub-sections:

A. Search & Refine ALR (Block A in the middle),
B. Assist & recommend ALR (Block B at the top-right),
C. Discover ALR Knowledge (Block C at the bottom),
D. Semantic Metadata Enrichments Software Ecosystem – SMESE V3; see (Brisebois, Abran, Nadembega, et al., Unpublished results). (top-left in Figure A 3.3 – see also Figure A 3.8).

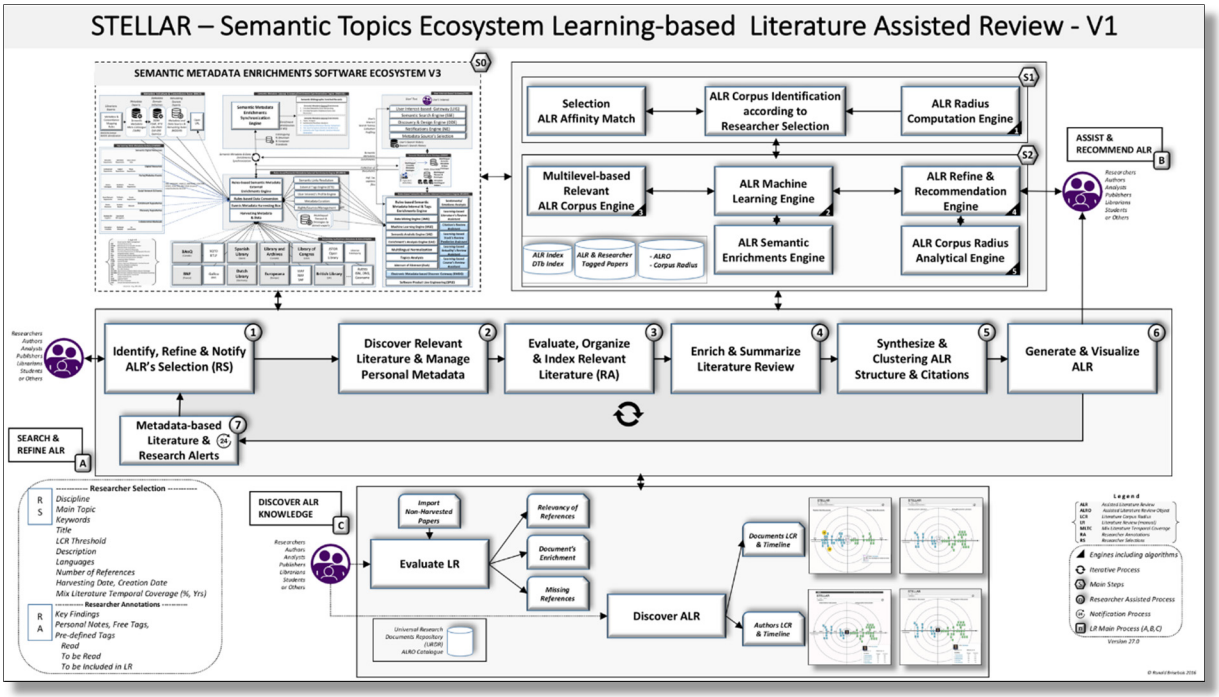


Figure A 3.3 STELLAR – Semantic Topics Ecosystem Learning-based Literature Assisted Review

3.3 SEARCH & REFINE ALR — Block A of the STELLAR prototype

The Search & Refine ALR (block A in Figure A 3.3) consists of seven steps – see Figure A 3.4:

1. **Identify, Refine & Notify ALR's Selection**

   This first step identifies and refines, in an interactive process, researcher selection (RS) metadata (i.e., documents selection parameters) in order to provide an ALR that meets researcher requirements; it also notifies the researcher when new paper which matches with its RS metadata is published.

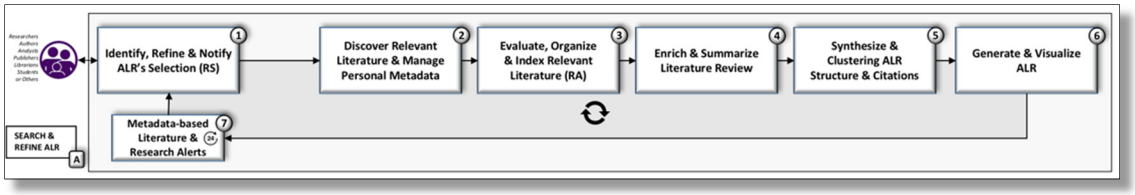


Figure A 3.4 Search & Refine ALR (Block A in Figure A 3.3)

A secondary objective of this step is to formulate the research questions. The metadata used to identify an RS are defined in two sections – see Table A 3.2:

a. Document Common Metadata section (top part of Table A 3.2),
b. Researcher Annotations section (bottom part of Table A 3.2).

The researcher can iterate this first step as necessary to complete the ALR or when there is a new paper to be added. Note that the papers are harvested in a master catalogue of papers defined in SMESE V3.

2. Discover Relevant Literature & Manage Personal Metadata

From the growing cluster of papers in SMESE V3, – a literature corpus that meets the RS metadata is identified. Any papers tagged by the researcher as "Relevant for the ALR" will be included. The paper relevancy is measured thanks to dynamic topic based index (DTb index) that is computed making used of TDM and MLM approaches.

3. Evaluate, Organize & Index the Relevant Literature

A subset of relevant papers is created in order to define the ALR Corpus based on the literature corpus radius index (LCR index). In contrast to Literature Corpus which denotes all the papers of a specific research topic, the ALR Corpus denotes only the papers of a Literature Corpus which meets RS metadata for an ALR. In other words, ALR Corpus is a subset of Literature Corpus in the same specific research topic.

4. Enrich & Summarize the Literature Review

The ALRO is produced through text summarization and subject extraction.

Table A 3.2 Researcher selection (RS) metadata

Number	Metadata	Description
A. Document Common Metadata		
1	Discipline	Selection of the discipline related to the ALR
2	Main Topic	The main topic is one of the most important metadata for building the ALR. It should be as specific as possible.
3	Literature Corpus Radius	The Literature Corpus Radius (LCR) is used to build other algorithms; it is the main concept that makes it possible to refine the selection of research documents to be included in the ALR.
4	Keywords	The researcher has to identify keywords representative of the ALR.
5	Harvesting Date	Date of document harvesting
6	Creation Date	Date of document creation
7	Title	Title of the ALR
8	MLTC - Mix of the Literature Temporal Coverage (Yrs, %)	The MLTC is very crucial to building and refining the ALR. It has two indicators: 1 - Number of years covered by the search 2 - Percentage of documents outside this time range to be included. Example: *When a researcher selects 5 years and 10%, STELLAR will select relevant documents published in the past five years and will include only 10% of documents falling outside this range.*
9	Description	A brief description of the research project of the ALR such as a paper abstract
10	Languages	The researcher has to choose the language of the documents to be included in the corpus of interest.
11	Number of References	The number of references that the ALR should consider.
B. Researcher Annotations Metadata		
12	Key Findings	The Key Findings are annotations regarding important findings in the document identified by the researcher.
13	Free Tags	The researcher may place tags on a document in order to remember some information about it. These tags can be used by STELLAR or the researcher to enhance the quality of the ALR.
14	Personal Notes	The researcher may attach notes to a document in order to remember some information about it. These notes can be used by STELLAR or the researcher to help specify the targeted ALR. Personal notes can be used a. to identify and understand the main points of a text b. to facilitate recall c. in later research and writing d. to make connections between different sources e. to facilitate rearranging the information for writing
15	Pre-defined Tags	These are predefined metadata to help the researcher and STELLAR track the status of the relevant document. Examples of pre-defined tags: a. Read b. To be read c. To be included in the ALR

5. Synthesize & Clusterize the ALR Structure & Citations

All the relevant documents are synthesized and organized into clusters related to the LCR index. This is done by putting the enrichments together in the ALRO pre-defined structure.

6. Generate & Visualize the ALR

In this step, the recommended papers in the Literature Corpus are generated and visualized. Assisted generation of the recommended papers helps the researcher examine the coherence of the ALR and iterate the ALR process. At any moment, the researcher can add to the relevant papers list that will be part of the final ALR.

7. Metadata-based Literature & Research Alerts

New relevant papers or new metadata related to the ALR are detected in this last step.

3.4 ASSIST & RECOMMEND ALR – Block B of the STELLAR prototype

Assist & recommend ALR (Block B in Figure A 3.3) allows refining the ALR through two sets of steps (S1 and S2) – see Figure A 3.5. Numbers 1 to 5 in the bottom-right corner of many of the boxes in Figure A 3.5 denote the MLM designed to identify a specific corpus, evaluate document relevancy or define learning models that are required by STELLAR for obtaining the ALR objects.

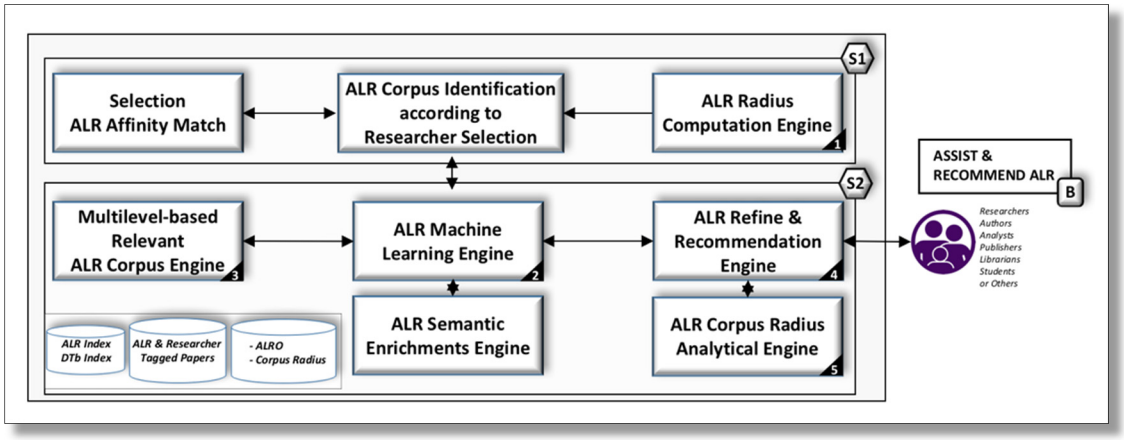


Figure A 3.5 Assist & recommend ALR (Block B in Figure A 3.3)

The ALR assistance and recommendation is done through TDM and MLM implemented in five algorithms. These algorithms refine the relevant literature candidates to build the final list of papers of the ALR:

S1 set of steps:

This set of steps identifies the papers that semantically matches the researcher selection (RS), taking the researcher annotations (RAs) into consideration as well. It includes:

- ALR Radius Computation of the LCR based on the metadata of the RS. This allows computing the LCR index of each paper of Literature Corpus making used of certain RS metadata;
- ALR Corpus Identification according to the RS: a semantic affinity match is applied considering LCR index to identify the ALR Corpus according to both the RSs and the RAs metadata. More details about this step are presented in Section 4;
- Selection ALR Affinity Match: the papers within the URDR whose metadata match the RS and RA parameters are identified; for example, the language of paper should match the RS language metadata.

S2 set of steps:

This set of steps S2 introduces the MLM 2 to 5 of the STELLAR prototype (more details in Section 4).

- **ALR Radius Analytical - MLM 5**
  All references related to the selected documents are identified and evaluated.

- **Multilevel-based Relevant LR Corpus - MLM 3**
  Creation of a dynamic list of relevant documents for building the ALR according to the RS. This process is dynamic: any new relevant research document may change the list of papers for building the ALR.

- **ALR Semantic Enrichments TDM**
  Enrichments are built from all the papers retained for the ALR. The enrichments are at different levels and are provided by the SMESE V3 platform: extraction of topics from the

documents, summarization of documents, and papers that refer to the papers retained for the ALR.

- **ALR Machine Learning - MLM 2**
  This step feeds the multilevel-based relevant ALR Corpus making use of DTb index and LCR index, for example by defining and creating the learning models used in the subsequent steps. More details are given in Section 4.2.

- **ALR Refine & Recommendation - MLM 4**
  This is the most important step for the researcher. It allows the researcher to refine all choices in terms of selections for building the ALR. The researcher is also presented with a number of recommendations for improving the ALR.

The following sources are used to build the suggested list of ALR papers:

1. The list of papers generated by the step 'ALR Refine & Recommendation - MLM 4' according to the RS; they are located in the centers of the circles in Figure A 3.6. This list includes the LCR threshold indicated by the gray circle (papers in blue);
2. The annotated papers from the researcher (RAs) – papers in red;
3. The papers identified by the Mix Literature Temporal Coverage (MLTC) from the RS – papers in yellow;
4. The universal research document repository (URDR), in the bottom right corner of Figure A 3.6, extracted from SMESE V3 (Brisebois, Abran, Nadembega, et al., Unpublished results).

Each corpus in Figure A 3.6 is shown as a circle whose horizontal axis represents the LCR line. Note that the origin of this axis is not explicitly visible. Indeed, the center of each circle denotes the origin of the horizontal axis going off toward the right or left, but the center is hidden by the type of metadata (RS or RA) used to select the corpus. However, here the direction (i.e., toward the right or the left) is not important. What is more important is to position a paper at the correct distance from the center according to its LCR index. The LCR index of a paper is defined as the similarity between the RS metadata and that paper's metadata such as title, topics, abstract and keywords. It measures the semantic relevancy of a paper

according to the RS. Note that, a paper on the right side is equal, in terms of meeting the RS metadata, to another on the left side at the same distance from the center.

The Literature Corpus contains all the papers regardless of their LCR index and the type of selection metadata (i.e., RSs or RAs). The papers within corpus radius are those located at the surface (forming a disc) of a circle with the specific corpus radius. We refer to the radius of this specific circle as the Corpus Radius (see Figure A 3.6).

Based on the definitions above, the Corpus Radius may be defined as the delimiter of the Literature Corpus suggested to the researcher for the ALR on the basis of the researcher's selections and annotations. The goal is to start from the entire Literature Corpus (i.e., the URDR) and use the selection process based on RSs and RAs to limit the number of papers to those that are relevant (recommended by MLM and tagged by the researcher). To facilitate understanding, both the RS and RA selection criteria are defined in the figure. The RS selection criteria are the researcher's metadata parameters while the RA selection criteria consist of notes, tags and key findings mentioned by the researcher.

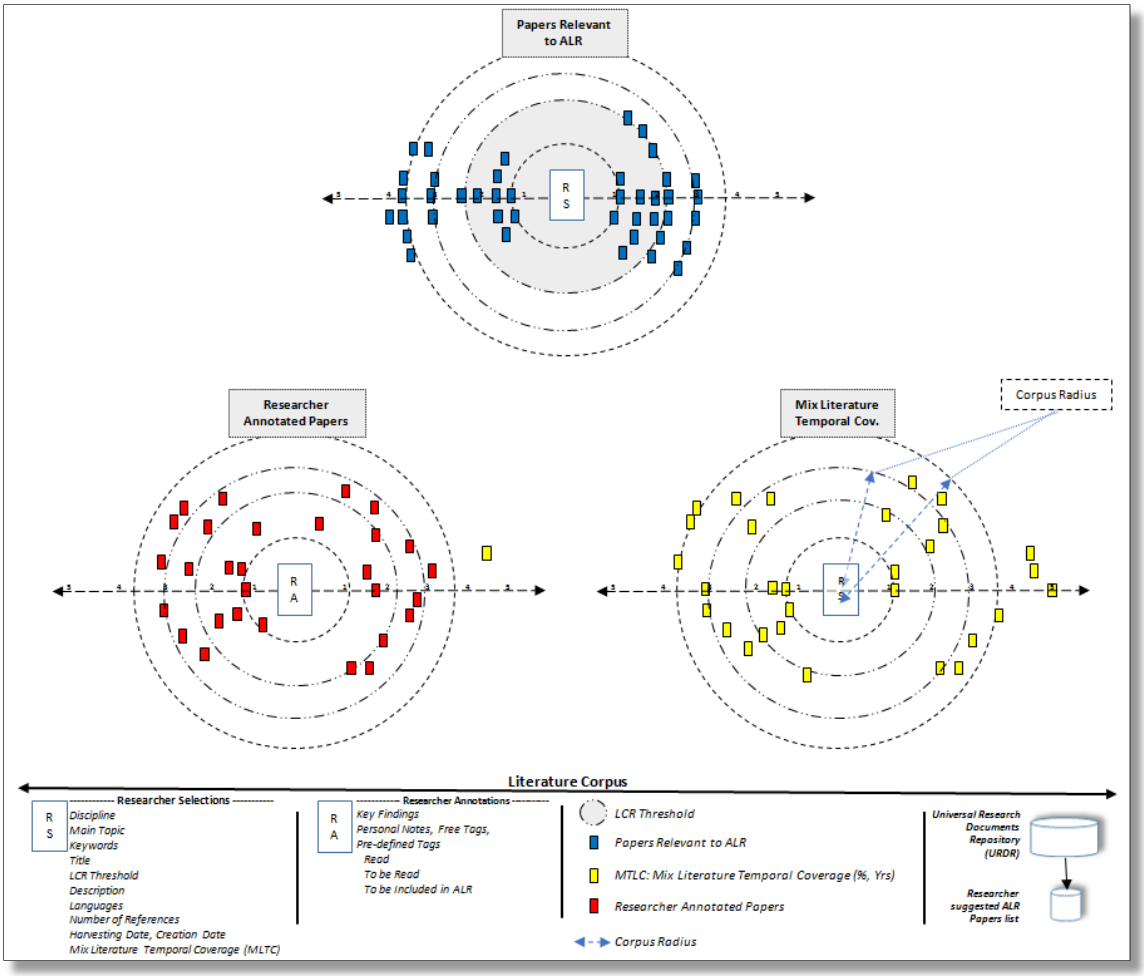


Figure A 3.6 Sources used to build the suggested list of ALR papers

To illustrate, consider the papers in the corpus radius called "***Papers relevant to ALR***" (disk with blue dots at the top of Figure A 3.6): all the papers within the gray disc are URDR papers whose LCR index is less than or equal to 2; in this case, the LCR threshold is set at 2.

3.5 Discover ALR Knowledge – Block C of the STELLAR prototype

The 'Discover ALR Knowledge' (Block C in Figure A 3.3) unveils the content of the ALR and checks the relevance of papers used to build a manual LR– see Figure A 3.7. It enables the researcher to explore the ALR information generated by STELLAR. As shown in Figure A 3.7, 'Discover ALR Knowledge' consists of two features:

1. Evaluation of manual LR that allows:
   a. Identifying the relevancy of manual LR references;
   b. Detecting missing references; in other words, the papers which should have been cited in the manual LR references.

2. Discover ALR feature includes:
   c. Graphical views of documents LCR and timeline,
   d. Graphical views of authors LCR and timeline.

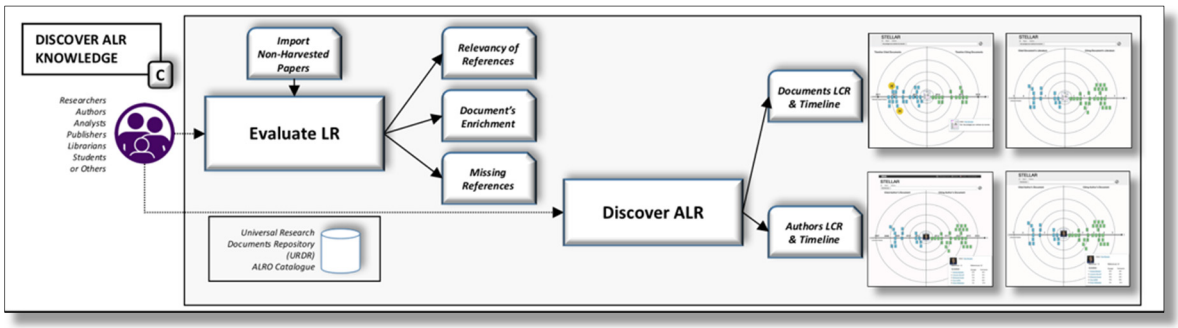


Figure A 3.7 Discover ALR Knowledge

More specifically, the first feature "Evaluate LR" consists in an assisted evaluation of an already published LR. This can be useful to researchers, students and teachers, helping them produce a better ALR related to their topic. To evaluate an existing LR, this feature compares the existing LR (done manually) to the one from STELLAR's MLM to quantify their similarity.

The second feature "Discovery ALR" consists in identifying the relative contribution of an author to a specific topic or area of interest. The contribution could be from different sources but the reputation of the journal has to be taken into account. Here are some examples of types of publications:

1. Papers in refereed journals,
2. Papers published online but subject to a rigorous review,
3. Books incorporating original research and published by reputable presses.

Here, the computation of the weight of a journal is not based on the number of papers it has published but on the number of papers it has published in the Corpus of papers (i.e., a collection of papers) defined by the researcher selection (e.g. the ALR Corpus).

The tags created by the researchers are used to enrich the ALR metadata. The process 'Discover ALR Knowledge' makes it possible to drill down through different types of visualization of the corpus, such as documents, authors and ALROs.

3.6 Semantic Metadata Enrichments Software Ecosystem SMESE V3 of STELLAR

The SMESE V3 platform presented in Figure A 3.8 (Brisebois, Abran, Nadembega, et al., Unpublished results) is a semantic metadata enrichment software ecosystem based on a multi-platform universal metadata model. It aggregates and enriches metadata to create a semantic master metadata catalogue (SMMC). This ecosystem consists of nine sub-systems:

1. Metadata initiatives & concordance rules,
2. Harvesting of web metadata & data,
3. Harvesting of authority's metadata & data,
4. Rule-based semantic metadata external enrichments,
5. Rule-based semantic metadata internal enrichments,
6. Semantic metadata external & internal enrichment synchronization,
7. Researcher interest-based gateway,
8. Semantic metadata master catalogue.

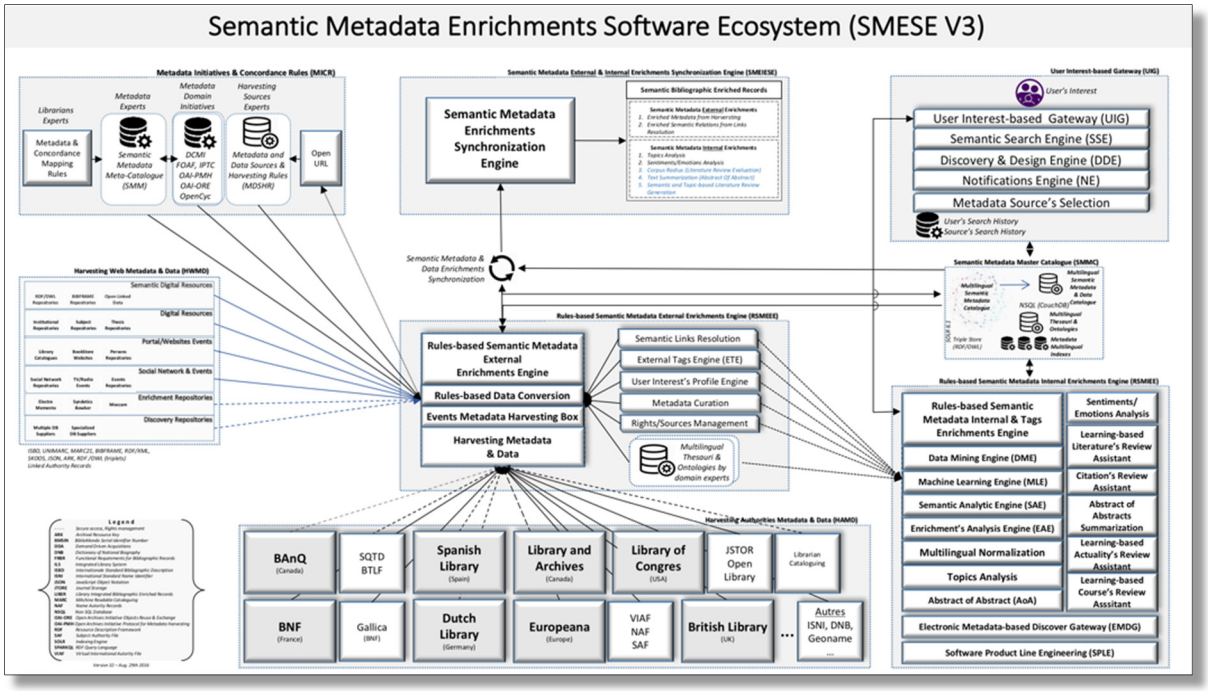


Figure A 3.8 SMESE V3 - Semantic Metadata Enrichments Software Ecosystem

The SMESE V3 platform allows enrichment from different sources including linked open data. Linked data is about using the Web to enrich related data or metadata by connecting pieces of data, information and knowledge on the Semantic Web.

SMESE V3 is essential to STELLAR for building its URDR (its base repository of harvested available papers at a given time t). This repository is growing every day and is required to notify the researcher of new relevant papers that may be used in the ALR.

3.7 Assisted Literature Review Object (ALRO)

The concept of the assisted literature review object (ALRO) is useful for managing ALRs. It is basically a component type that includes many types of information related to the LR. Indeed, many kinds of information can be useful in building the ALR, for example:

1. Researcher annotations (RAs),
2. Metadata sets,
3. Datasets,

4. Slide presentations,
5. Research reports,
6. Hypotheses investigated during the research,
7. Results produced from prototypes,
8. Unique identifiers.

In Figure A 3.9, the Entity Matrix has been modified with the addition of a new component type: ALRO (Bechhofer et al., 2013). An ALRO aggregates all objects and relationships related to the creation of an ALR. All this information can be re-used in subsequent research investigations. An ALRO can be also identified by a uniform resource identifier (URI) such as the digital object identifier (DOI). An ALRO can be shared by researchers or re-used to accelerate research findings.

In addition, each type of text has its own specific structure. Scientific articles are often organized as follows:

1. Abstract,
2. Introduction,
3. Problem description,
4. Research questions,
5. Literature Review or Related Literature or Related Work,
6. Methodology,
7. Key findings (results),
8. Conclusions,
9. References.

The algorithms used to perform ATS for scientific papers need to take this text organization into account. To be able to generate an ALRO, STELLAR proposes an ALR template:

1. Title,
2. Abstract of Abstracts (AoA),
3. Keywords,
4. Literature Review Summary,
5. References,

6. Researcher Selection.

STELLAR proposes different types of ALRO index to evaluate the relevance and importance of am ALRO for a specific researcher; for example, the DTb index of an ALRO in STELLAR takes into account:

1. Topic-based approach,
2. Text-based approach,
3. Reference-based approach,
4. Author-level metrics,
5. Co-author-level metrics,
6. Venue-level metrics,
7. Social-level metrics,
8. Affiliation-level metrics.

The ALRO metadata (see Figure A 3.9) are the basis for the identification and indexing of a specific ALRO. Typically, the metadata of an ALRO include:

1. Venue,
2. Title,
3. Abstract,
4. Authors,
5. Issue of publication,
6. Volume of publication,
7. Publisher,
8. Page numbers,
9. Date of publication,
10. ISBN,
11. DOI,
12. ISSN,
13. Keywords,
14. Annotations.

ENTITY (NOTICES) MATRIX of the SMESE's Master Catalogue (EXAMPLE)

Calendars
- *Interests*
- *Library*
- *POI*
- *Rewards*
- *TV Channel*
- ...

Collections
- *Interests*
- *Library*
- *Organizations*
- *Personal*
- ...

Contents
- *Audio Books*
- *Books*
- *Cartographic Mat.*
- *Citations*
- *Comics*
- *Estampes*
- *Manga*
- *Microforms*
- *Movies (DVD)*
- *Music (CD)*
- *Musical Partitions*
- *Old Books*
- *Photos (Image)*
- *Press*
- *Serials*
- *Sounds*
- *Videos*
- ...

Documents
- *Google Doc*
- *Paint*
- *PDF*
- *Powerpoint*
- *Spreadsheet*
- *Word*
- ...

Events
- *Cinemas Rep.*
- *Expositions*
- *Libër Spirits*
- *News*
- *Notifications*
- *Press Conference*
- *Shows*
- *Spectacles*
- *Theaters*
- *TV Shows*
- ...

- ***ALRO***
- *Titre*
- *Topics*
- *Keywords*
- *#references*
- *Annotations*
- ...

Objects
- Object
- Work of Art
- ...

Persons
- *Actor*
- *Author*
- *Celebrity*
- *Musician*
- *Politician*
- *Producer*
- *Singer*
- *Students*
- *User*
- ...

Places
- *City*
- *Localization*
- *POI*
- ...

Products
- *Financial*
- *Groceries*
- *Hardware*
- *Natural Health*
- *Pharmacy*
- *Software*
- ...

Publications
- *Articles*
- *Education Programs*
- *Fact Sheets*
- *Questions/Answers*
- *Manuals*
- *Monographs*
- *Newsletters*
- *PostCards*
- *Posters*
- *Proceedings*
- *Thesis*
- ...

Rewards
- *Litterature*
- *Movies*
- *Music*
- *Nobel*
- ...

Resources
- *Online*
- *Physical*
- ...

Subjects
- *Genomes*
- *MindMaps*
- *Ontologies*
- ...

WebSites
- *Homework Help*
- *Youth*
- ...

Works
- *Concepts*
- *Expressions*
- *Manifestations*
- ...

Figure A 3.9 Entity matrix of the SMESE V3 Platform Master Catalogue

In STELLAR, additional metadata are included and classified into three categories (see Table A 3.3):

1. Document metadata,
2. Researcher metadata,
3. Author metadata.

Table A 3.3 STELLAR additional metadata

Document metadata	Researcher metadata	Author metadata
Domain	FreeTags	SearchFields
Language	Notes	Awards
Citations with category	KeyFindings	Affiliated institution
References	Tags	Co-authors
Citing_documents		Courses
Section		NumberOfPublication1stAuthor
Figures		NumberOfPublication2ndAuthor
Tables		NumberOfPublicationOther
Rights		NumberOfGraduatedStudentPhD
		NumberOfGraduatedStudentMaster

Several supervised MLM-based metadata extraction methods are available for automatic integration of metadata into bibliographic manager tools such as Endnote. In this work, which takes a rules-based approach, a supervised MLM is used (Gulo et al., 2015). The metadata are extracted from databases such as www.opendoar.org, www.researchgate.net, www.academia.edu, and OAI-PMH sources.

Additional metadata about authors and researchers need to be identified or computed. Author metadata is usually the basis of a search for document relevancy detection. They help to gain insights about author' publications.

4. STELLAR Processes Description

This section presents the MLM of STELLAR. For an improved understanding of Steps 1 and 2 of STELLAR (as indicated in Figure A 3.3), Figure A 3.10 presents an overview of the STELLAR processes, their inputs and outputs and their interoperability. Each one of these five STELLAR processes is described in more detail in the following sub-sections.

From now on in this paper, the following terms are used interchangeably: document, paper and scientific paper.

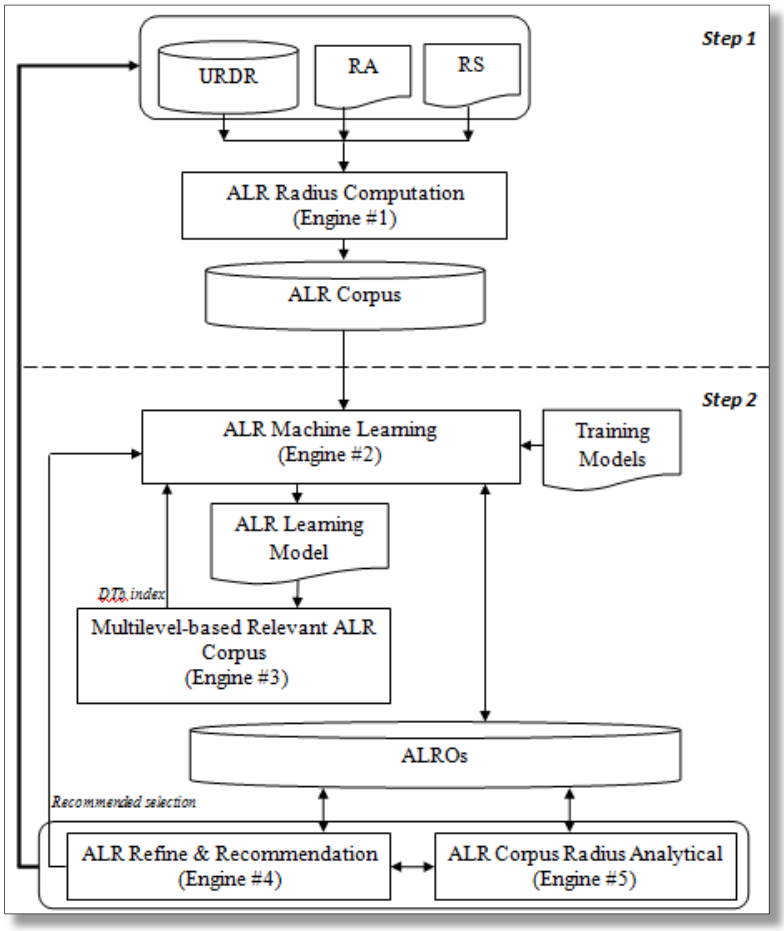


Figure A 3.10 Interoperability of the STELLAR processes

1. Using as inputs the URDR that contains existing ALROs, as well as papers, RAs and RS, the ALR radius computation engine computes the LCR index. The LCR index is then used by the ALR Corpus identification engine in addition to selection affinity match (see Figure A 3.3) to generate an ALR Corpus that meets the researcher's requirements (i.e., RS and RAs);
2. Next, using as inputs the ALR Corpus and the training models built by selected researchers, MLM provide the ALR learning model used by the Multilevel-based Relevant ALR Corpus. MLM also enrich the ALR Corpus to provide the ALRO;

3. The Multilevel-based Relevant ALR Corpus computes the DTb-index that measures the relevancy of each paper in the ALR corpus;
4. Making use of the generated and enriched ALRO, the ALR Refine & Recommendation engine suggests the Paper References list to the researcher;
5. The ALR Radius Analytical generates different analytical views of the ALR Corpus.

4.1 ALR radius computation

ALR radius computation is used to rank the relevancy of papers to be included in the ALR, according to the researcher selection (RS) and researcher annotations (RAs). Computation of the LCR index is defined as a sub-algorithm of the semantic ALR selection search that identifies the ALR corpus according to the RS and RAs defined in Figure A 3.3. Here, selection metadata and selection parameters may be used interchangeable.

To identify an ALR corpus as shown in the Step 1 of Figure A 3.10, the selection parameters (RA and RS) are classified into three categories (see Table A 3.4):

1. Evaluation-based,
2. Selection-based,
3. Sort-based.

Table A 3.4 STELLAR classification of selection parameters

Evaluation-based	Selection-based	Sort-based
Main Topic (MaT)	Discipline	Literature Corpus Radius (LCR)
Keywords (KeW)	Languages	Mix of the Literature Temporal Coverage (MLTC)
Title (TiT)	Document Researcher Annotations	Number of References
Description (DeC)		

1. In evaluation-based selection, the LCR index is computed based on the TDM approach. This class of RS is mainly used in the ALR radius computation to evaluate the LCR index used by sort-based selections;
2. In selection-based selection, documents are selected based on a specific value of the document metadata. As shown in Figure A 3.11, in this class of parameters, the document's Researcher Annotations (RAs) are included and consist of:
   a. Key Findings,
   b. Free Tags,
   c. Personal Notes,
   d. Pre-defined Tags.

3. In sort-based selection, a specified number of documents are sorted according to a particular order. For example, for an ALR in a given field, the researcher may need to keep:
   a. Z% of relevant documents that are X years old or less, and
   b. (100-Z)% that are more than X years old.

Figure A 3.11 illustrates the interaction between the researcher selections. To allow researchers to combine the selection parameters themselves according to their experience in order to obtain a corpus that meets their requirements, an option for selection condition formatting is available through the "***Researcher search experience***" function – see leftmost box in Figure A 3.11.

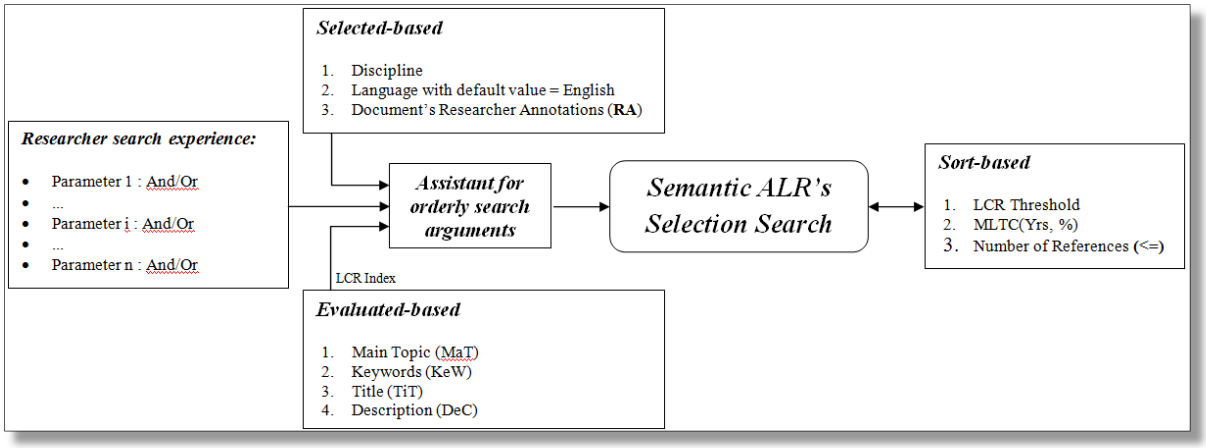


Figure A 3.11 Researcher selection and annotations

For example, Figure A 3.12 shows the steps (A to D) in a semantic ALR selection search for the more complex case of a selection condition based on RS and RA: "*Discipline* AND *Language* AND *RA-(To be included in the ALR)* AND *LCR Threshold* AND *MLTC* AND *Number of references*.

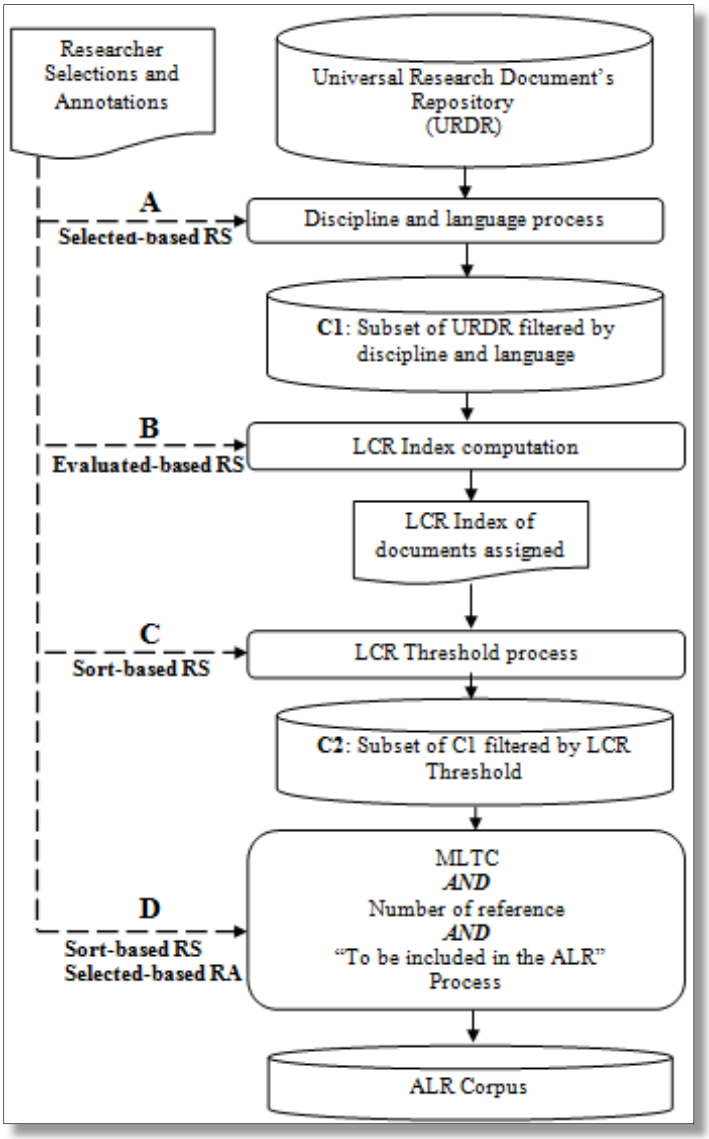


Figure A 3.12 Steps in a semantic ALR selection search

In the following paragraphs, the TDM semantic topic search for the example of Figure A 3.12 is explained in detail.

A. Discipline and language researcher selections step

In step A in Figure A 3.12, the volume of documents to be considered for the rest of the process may be reduced, based on:

1. Discipline selection: selecting all documents that are in the Meta Corpus of a given discipline, e.g., Biology and Computer Science;
2. Language selection: limiting the documents to be considered for the ALR to a specific language; the default value is English.

The selection query uses the document metadata in the URDR.

Let *DC* be the chosen discipline, let *LG* be the given language, let DISCIPLINE be the metadata that records the discipline of the documents in URDR, let LANGUAGE be the metadata that records the language of the documents in URDR and let *DiscLan_Corpus(DC,LG)* be the set of documents in the language ***LG*** that are in the discipline ***DC***.

DiscLan_Corpus(DC,LG) is obtained as follows:

*DiscLan_Corpus(DC, LG) = [**select in** URDR the Documents **where***
DISCIPLINE is "DC"
and
LANGUAGE is "LG"]

This query to the URDR extracts only papers in the specified discipline and language.

Let C_1 be the corpus of papers obtained in step A.

B. LCR index computation step

Using the set of papers extracted in step A, the LCR index is computed next in step B based on the evaluation-based selections: main topic, keyword, title and description.

The impact of each of these selections is computed to identify the papers that best match the researcher selections:

1. First, the similarity matching of each evaluation-based selection with a predefined selection of papers is evaluated within the range [0,1]: 1 means the most similar while 0 means the least similar;

2. Next, based on their predefined weight and the similarity matching value, the LCR index is computed.

The LCR index computation step consists of five sub-steps, a to e. Appendix A presents the details of all the algorithms used.

a. Similarity matching of researcher main topic with topics extracted from document abstracts

The similarity matching of the researcher main topic with the topics extracted from the document abstracts is first computed using the topic detection ML model called BM-Scalable Annotation-based Topic Detection (BM-SATD) (Brisebois, Abran, Nadembega, et al., Unpublished results). More specifically, BM-SATD uses multiple relations within a term graph and detects topics from the graph using a graph analytical method. BM-SATD combines semantic relations between terms with co-occurrence relations across the document, by making use of the document annotations.

Here, the similarity matching is based on the n-gram approach where the value n is used as the weight (Bertin, Atanassova, Sugimoto, & Lariviere, 2016): when the i-gram expression of the researcher main topic is found in the abstract, the weight i is associated with this expression (see equations A.1 to A.3 in Appendix A).

b. Similarity matching of researcher keywords with document keywords

The similarity matching of the researcher keywords is computed next by making use of the KEYWORDS sections of the documents. The impact value is the number of researcher selection keywords that are similar to the KEYWORDS section (see equations A.4 and A.5 in Appendix A).

c. Similarity matching of researcher title with document titles

Before this similarity matching computation, the researcher title and document titles are pre-processed to filter noise. This consists in stemming, phrase extraction, part-of-speech filtering and removal of stop-words. Next, based on the terms obtained, the maximum n-gram of the

researcher title which is met in the document title is used as the title selection impact value (see equations A.6 and A.7 in Appendix A).

d. Similarity matching of researcher research topic description with document abstracts

The researcher research topic description is semantically compared with the document abstract in order to measure the semantic similarity level. This similarity matching makes use of WordNet::Similarity (Pedersen et al., 2004), which applies six measures of similarity and three measures of relatedness; thus, several terms may be semantically the same. To measure this similarity, the TF-IDF approach is extended to meet our objective by applying it to the vocabulary of the corpus instead of the document itself (see equations A.8 to A.10 in Appendix A).

e. LCR index computation

Finally, when the similarity matching of each evaluation-based selection has been completed through sub-steps a to d, the LCR index within the [0,1] range can be computed. Note that the LCR index is a weighted sum of the computed value of each evaluation-based selection.

The difference in weight between two consecutive evaluation-based selections (i.e., selection i and selection i+1) is a predefined constant value (see equation A.11 in Appendix A).

C. Literature Corpus Radius (LCR) threshold selection step

In this step, a set of documents is sorted or selected according LCR index value. For example, a researcher may indicate that the LCR threshold is 0.7; the output will then be a subset of corpus C whose LCR index is greater than or equal to 0.7. When the researcher does not give this selection, the set of documents obtained in step A above (Discipline and language researcher selections) is used as the input of this step.

Let C_2 be the corpus of documents obtained in step C.

D. MLTC AND Number of references AND "To be included in the ALR" step

MLTC is the Mix Literature Temporal Coverage. Let MLTC (x, y) with its number of selections equal N: this means the researcher expects to have at most ***N*** documents, with a maximum of ***(100-x)%*** (i.e., $\frac{N}{100} \times (100 - x)$) that are at most ***y*** years old, and including all the documents tagged "To be included in the ALR". Note that the latter documents have priority.

First, a list (in descending order) is created based on the LCR index applied to corpus C_1 where the documents tagged "To be included in the ALR" are at the top due to their priority.

Let All_C_1 be this list. New_C_1 is defined as a sub-list of C_1 in which the document age is less than or equal to *y*, and Old_C_1 contains documents older than *y*.

Let $A = \frac{N}{100} \times x$ be the length of New_C_1 and $B = \frac{N}{100} \times (100 - x)$ be the length of Old_C_1. To take into account the three selections made in sub-step D, a pseudo-code is proposed in Appendix B.

Note that, when the number of documents in All_C_1 is less than N, all the documents are considered affinity matches for the ALR; in that case, the MLTC selection is ignored.

However, when there are not enough documents whose age is less than or equal to *y* to satisfy the MLTC selection, a new MLTC is provided in order to reach the number *A*. But if the researcher requires the MLTC selection to be met, some documents are removed from New_C_1 in order to meet the selected MLTC(x, y).

If an "OR" has been placed between the researcher selections, the LR corpus will be defined as the union of the C2 subsets provided by the MLTC process, the Number of references process and the "To be included in the ALR" tags.

4.2 ALR Machine Learning (ALRML)

ALR Machine Learning (ALRML) (Step 2 of Figure A 3.10) for semantic ALR selection is the core of STELLAR. It is the only process that interacts with all the algorithms of the other MLM, combining the TDM and MLM approaches to discover hidden information in papers. This information is used as internal semantic enriched metadata.

ALRML is a supervised MLM that makes use of a training set in order to provide the learning model, called the ALR learning model, composed of three sub-models:

1. Section recognition learning model,
2. Citation-based learning model,
3. Text-based learning model.

For the rest of this sub-section, the following two expressions are used:

1. Cited document: denotes the paper cited by another paper,
2. Citing document: denotes the paper citing another paper.

4.2.1 Section recognition learning model

Unlike most other types of documents, scientific papers present similarities in terms of structural organization, with common sections as follows:

1. Abstract,
2. Introduction,
3. Related work,
4. Methodology,
5. Results,
6. Discussion,
7. Conclusion,
8. References,
9. Appendices.

The section recognition learning model in STELLAR supports the assumption that knowing the section in which a sentence appears may change its context. For example, citations in the 'Related Work' section do not carry the same weight as those in the 'Discussion' section in terms of identifying existing papers in a specific domain. In STELLAR, the following sections are considered: abstract, introduction, literature review, solution or methodology, results, and conclusion.

To initialize the learning model, the section titles are classified on the basis of the training set. In addition, different scenarios of structural organization have been observed. For example:

1. The main scenario is: (abstract, introduction, literature review, solution, results, and conclusion) or (abstract, introduction, solution, results, and conclusion);
2. A second scenario is that the ALR is included in the 'introduction' section.

In both scenarios, the abstract and introduction are first and the conclusion last. Table A 3.5 provides an example for each section. To refine this learning model, the semantic similarities are computed based on a manual titles classification (i.e., titles found by humans) and the WordNet lexical database. For the manual classification, researchers are selected from the URDR are selected and asked to read and label the section headings of selected papers; this generates the section recognition training model incorporated into the "Training Model" mentioned in Figure A 3.10. To enrich the learning model, when a section heading is detected in a document but is not mentioned in the current section recognition learning model, it can be placed in the right category through the semantic similarity process.

Table A 3.5 Commonly used section headings in scientific papers

Section label	Section headings	
	Manually detected	**Automatically detected**
Abstract	Abstract	-
Introduction	Introduction	-
Literature review	Literature review, related work	Background, previous work, related literature, existing approaches
Solution	System model, proposal model	Proposed system, design, the system, methodology
Results	Results, experimentation, simulation, experimental, empirical	Experimental results, implementation, evaluation, discussion, implementation details, experimental setup
Conclusion	Conclusion, conclusion and future work	-

4.2.2 Citations-based learning model

A citations-based learning model has been designed to identify and extract citations in documents. This learning model is divided as follows (see Table A 3.6):

A. A citation style learning model based on citation style;
B. A citation classification learning model based on citation rhetorical categories and cue phrases.

Table A 3.6 Citations-based learning model

A. Citation style learning model	
Style marker	**Description**
Numerical marker	The syntax of this citation style is the number between brackets; for example, [1 to N] where N is the total number of references.
Textual marker	There are two syntaxes for this citation style: (<names of authors>, year) or < names of authors > (year).
Personalization marker	This style is based on the set of texts that refer to cited papers. After the numerical and textual markers, the cited document is referred to by the author's name or a personal pronoun. The name of the proposed solution or algorithm may also be used to refer to a cited paper.
B. Citation classification model	
Citation category	**Description**
Relevant	According to the citing document, the cited document is relevant for the domain.
Problem	The cited document presents the issues that led to the research.
Uses	The cited document proposes a solution that is used in the citing document.
Extension	The cited document proposes a solution that is extended by the citing document.
Comparison	The cited document proposes a solution that is compared with the citing document solution in terms of performance.

More specifically, the citation categories are identified based on rhetorical expressions detected through cue phrases. A cue phrase is the phrase that often occurs in a certain rhetorical category. In the case of citation classification, the verb plays the main role. For example, the verbs "proposed", "presented", "introduced" and "described" are used in rhetorical expressions in the Solution section. Researchers are asked to read and detect the cue phrases associated with each citation polarity (i.e., good opinion or bad opinion) and category; this makes it possible to build a training model of cue phrases and their classifications, which is integrated

into the "Training Model" mentioned in Figure A 3.10. This manual annotation is done before the STELLAR MLM process (see ALRML).

Next, based on semantic similarities, any rhetorical category that was not detected manually is detected automatically and added to the model. In addition to categories, the polarity model is proposed in order to indicate whether the citation is positive or negative.

The classification model consists of:

1. The citation polarity learning model, which contains a list of rhetorical expression polarities (PR);
2. The citation category learning model, which contains a list of rhetorical expression categories (CR).

4.2.3 Text-based learning model

To define the text-based learning model, text categories have been predefined as follows:

1. Problem,
2. Solution,
3. Results.

As in the citation-based learning model, rhetorical expressions are detected by means of cue phrases:

1. First, cue phrases that often appear in certain rhetorical expressions are manually identified;
2. Next, semantic similarity is applied automatically to these cue phrases in order to build the learning model. For example, "We", "This paper", "This article" and "In this paper" are often used with the verb "present", "propose" or "introduce" to present the solution. Here is an example of a rhetorical expression that presents the problem: "*Communication efficiency can be largely improved if the network anticipates the needs of its users on the move and, thus, performs reservation of radio resources at cells along the path to the destination*." The authors' solution is presented in the next sentence: "*In this vein, we propose a mobility prediction scheme for MNs; more*

specifically, we first apply probability and Dempster–Shafer processes for predicting the likelihood of the next destination, for an arbitrary user in an MN, based on user habits (e.g., frequently visited locations)."

The text-based learning model is organized as follows:

1. The cue phrase learning model containing a list of cue phrases (CPs):
   a. Problem CP,
   b. Solution CP,
   c. Result CP.

2. The thematic learning model, which contains a list of thematic rhetorical expressions (TRs):
   a. Problem learning model: list of problem rhetorical expressions (P_TR):
      - Context P_TR,
      - Limitation P_TR.
   b. Solution learning model: list of solution rhetorical expressions (S_TR):
      - Algorithm S_TR,
      - Concept S_TR,
      - Approach S_TR,
      - Technique S_TR.
   c. Result learning model: list of result rhetorical expressions (R_TR)
      - Outperformance R_TR,
      - Sub performance R_TR.

4.3 Multilevel-based relevant ALR Corpus

The multilevel-based relevant ALR Corpus (in Step 2 of Figure A 3.10) is presented here. It is used to evaluate the relevancy of a paper based on a number of scientometric measurements. Here, relevancy is not based on RAs and RS; instead, the input corpus used by the multilevel-based relevant ALR Corpus is the ALR Corpus obtained through the ALR's semantic search based on RAs and RS. The measurement of relevance is referred as the ALR Index.

Three types of ALR Index are defined in STELLAR:

1. Personal,
2. Collaborative,
3. Dynamic topic-based (DTb).

With the personal index, the ALR can be restricted to documents tagged by the researcher as "To be included in the ALR".

The collaborative index extends the personal index by including documents tagged "To be included in the ALR" by a specific community of researchers.

The dynamic topic-based index (DTb index) selects documents for the ALR when the researcher has not requested a personal or collaborative index. The DTb index is a weighted sum of the values that denote the importance of the different inputs considered, classified as:

1. Key findings and peer citations index,
2. Venue index,
3. Document references index,
4. Authors and their affiliated institutes.

Unlike existing approaches, the DTb index is not limited to journal-level metrics; it also considers conference proceedings and workshop metrics, and this makes it venue-level metric based.

Appendix C presents the details of the algorithms used to compute the ALR Index.

4.4 ALR Refine & Recommendation MLM

The ALR Refine & Recommendation MLM (in Step 2 of Figure A 3.10) is presented here. The input is the ALR Corpus of relevant and enriched papers identified automatically by STELLAR to recommend selections parameters to a researcher (see previous sections). This MLM may next recommend three different aspects of the ALR selection (Figure A 3.13):

1. The list of papers to be included in or removed from the ALR,

2. The number of references (i.e., papers) to be considered for the ALR,
3. The % of Mix Literature Temporal Coverage (MTLC) to be included in the list of references.

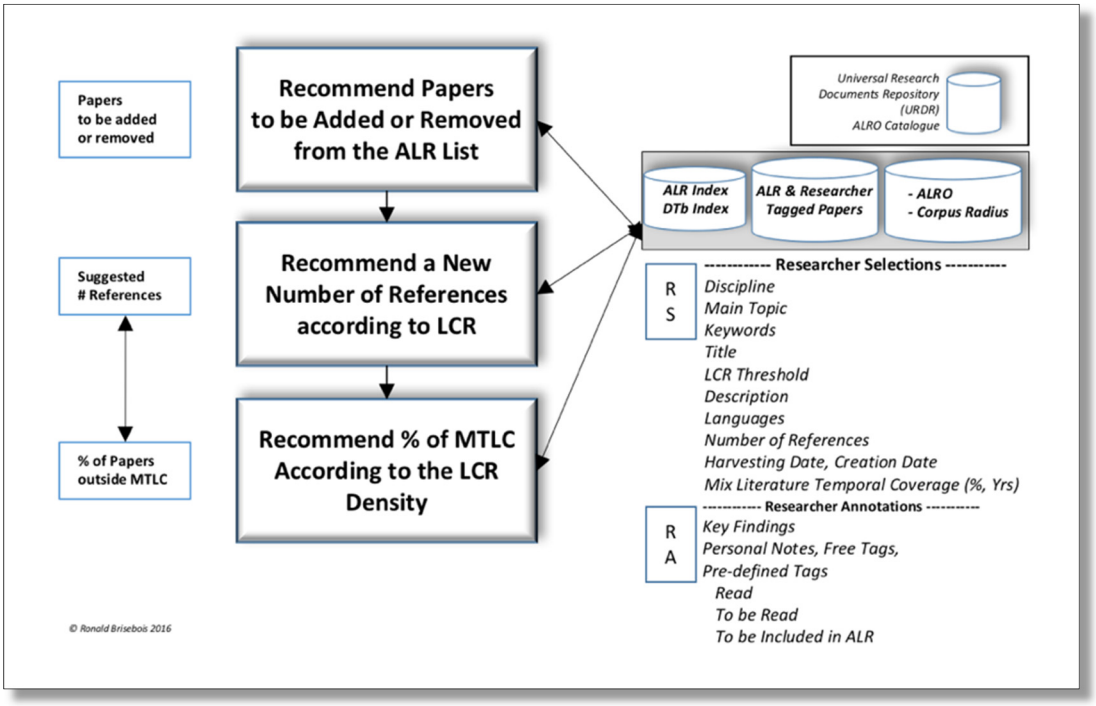


Figure A 3.13 Refinement & Recommendation MLM

To help the researcher to choose the right combination of parameters (RS), the refinement function makes recommendations in the following three areas:

1. Identification of documents to form the recommended list for the ALR:
   a. Launch the Multilevel-based Relevant ALR Corpus engine to actualize the proposed document list for the ALR with the default STELLAR options;
   b. Compare with the first list and recommend additions or removals.

2. Identification of the optimal number of documents as references to include in the ALRO. This recommendation is related to the LCR and based on the most relevant documents closest to the selected topic; the highest number will be the proposed number of references. The sub-steps are:

a. Launch the Multilevel-based Relevant ALR Corpus engine to actualize the list of documents proposed for the ALR with the default STELLAR options and the ALRO selection;
b. From the list of proposed documents, take the distribution of LCR and create a dataset;
c. Identify the number of references in the optimized dataset (i.e., the most relevant documents closest to the selected topic); this then becomes the recommended number of references;
d. The researcher is able to modify the number of references at any time to obtain a new recommendation.

3. Identification of the % of MTLC to be part of the ALR.
   a. Launch the Multilevel-based Relevant ALR Corpus engine to actualize the proposed document list for the ALR with the default STELLAR options and the ALRO selection;
   b. Based on the proposed list of documents included through the % of MTLC, take the distribution of LCR and create a dataset;
   c. Identify the % of MTLC in the optimized dataset; this then becomes the recommended %;
   d. The researcher is able to modify the % of MTLC at any time to obtain a new recommended %.

4.5 ALR Corpus Radius Analytics

The ALR Corpus Radius Analytics (in Step 2 of Figure A 3..10) is presented in this section: it presents a number of ways of viewing the list of documents for drill-down purposes. This sub-section describes the concepts used in producing an assisted ALR, including:

1. The Timeline of a Document-based Literature Corpus Radius,
2. The Literature Corpus Radius (LCR).

Two classes of documents are defined:

1. Citing documents,
2. Cited documents.

For a better understanding, let *d* be a considered document; a citing document is a document that cites document *d* while a cited document is a document that is cited by document *d.* The Figure A 3.14 illustrates the two classes of documents in reference to the publishing date.

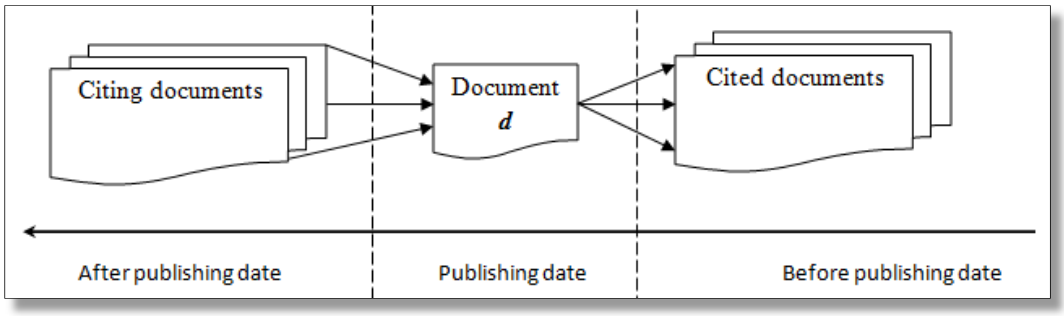


Figure A 3.14 Two classes of documents in reference to the publishing date

Figure A 3.15 shows a Timeline of a Document Corpus Radius, where the horizontal axis indicates the Literature Corpus Radius. The horizontal timeline indicates the range of publishing dates—in this example, from 2007 to 2011 and from 2012 to 2016.

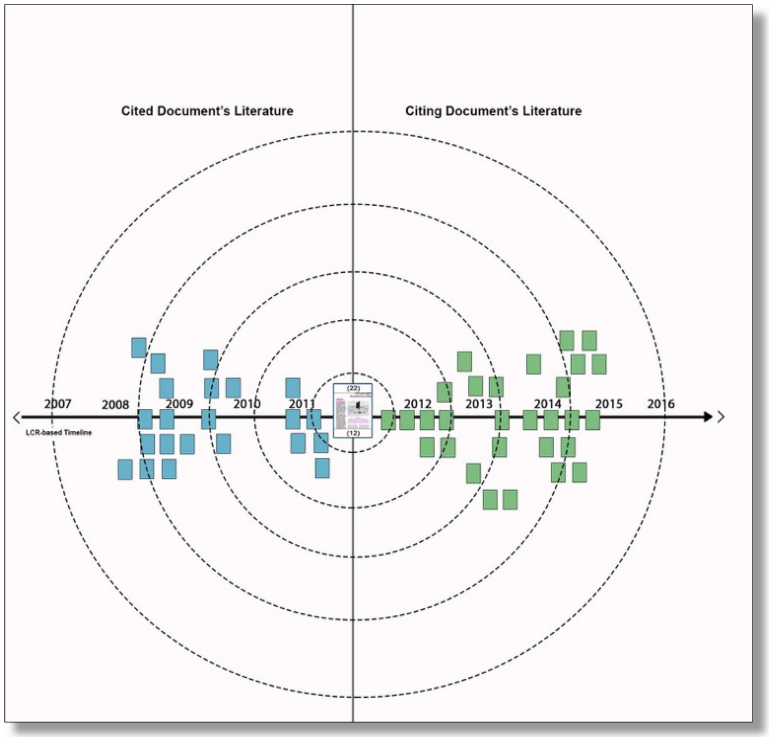


Figure A 3.15 Timeline of a Document-based Literature Corpus Radius

The radius is the distance from the center of the circle to the cited paper (left side) or to the citing paper (right side). It is thus a measure of the relevancy of a paper according the researcher selection of parameters.

Next, Figure A 3.16 presents the Document-based Literature Corpus Radius, with the horizontal axis indicating the LCR value (from 0 to 5). The closer a paper is to the center of the circle, the more relevant it is to the ALR.

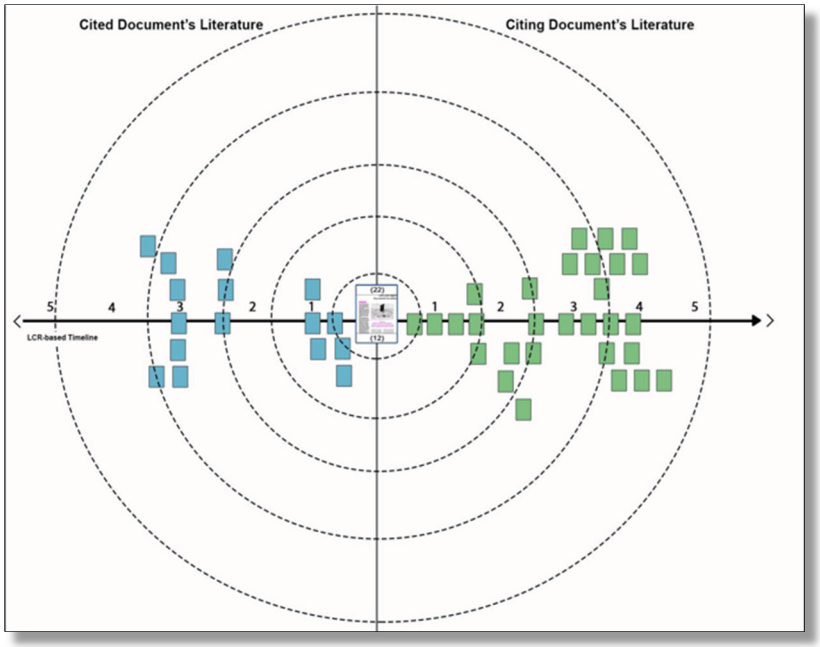


Figure A 3.16 Document-based Literature Corpus Radius

The radius denotes the temporal distance from the center document to the Cited Document's Literature (left side) or to the Citing Document's Literature (right side).

5 STELLAR Performance Evaluation Through Simulations

This section presents an evaluation of the performance of the STELLAR prototype through a number of simulations limited to the identification of relevant papers for an ALR.

5.1 Datasets

Two datasets were used for the simulations:

1. A dataset harvested from databases,
2. A baseline dataset.

5.1.1 Dataset harvested from databases

For the simulations, 2,000 scientific papers were collected from databases such as ScienceDirect and Scopus. The papers dealt with various research topics in Computer Science. Two sub-domains were chosen, each with 1,000 papers:

1. Artificial Intelligence,
2. Information Systems.

In the context of these simulations, the sub-domains are treated as domains. The other metadata were collected as bibliographic references.

For each paper, the downloaded bibliographic files were parsed to extract the metadata and were input into the SMESE V3 platform with the paper itself. Here, a scenario was defined as a set of two simulator runs, one on each domain dataset. For the simulator run parameters, the metadata of one paper in the dataset (discipline, language, title, topic, keywords and abstract) were used as the RS and RA parameters.

5.1.2 Baseline dataset

For the present study, we had already produced a manual ALR that included all the papers listed in our References section. This manually assembled list was used as the baseline dataset to evaluate the performance of the STELLAR prototype. The baseline dataset consisted of 58 papers dealing with both general and specific topics within the domain. Here, a scenario was defined as one simulator run where the 58 papers constituted the dataset. For the simulator run parameters, the metadata of the present study (discipline, language, title, topic, keywords and abstract) were used as the RS and RA parameters.

5.2 Performance criteria

The STELLAR prototype was evaluated from the viewpoint of its users: researchers, students, authors, publishers and librarians. As in (Rúbio & Gulo, 2016), two performance criteria were used to assess the relevancy of the papers for the researchers:

1. Accuracy: the percentage of true classifications,
2. Precision: the percentage of the classified items that are relevant.

Considering the sets of relevant papers (REL) and non-relevant papers, (NREL), true relevant (TR) denotes the papers classified as REL when they really are, while false relevant (FR) denote the papers classified as REL when they are not. Thus, with the same logic, the papers classified as NREL can be true non-relevant (TN) or false non-relevant (FN). For each type of dataset, the definition of a scenario is given in sections 5.1.1 and 5.1.2 according to the type of dataset.

Accuracy (denoted by a) and precision (denoted by p) were computed as follows for each scenario:

$$a = \frac{TR + TN}{TR + FR + TN + FN} \qquad p = \frac{TR}{TR + FR}$$

To identify TR, FR, TN and FN for each scenario, a target paper was chosen for the domain; next, the metadata of this target paper were used as the researcher selection parameters and the references papers in the output set of the prototypes were compared to the cited papers of the target paper. Through this comparison, TR, FR, TN and FN were defined.

Let $a_{i,j}$ be the accuracy of the scenario i^{th} of the dataset j and $p_{i,j}$ be the precision of the scenario i^{th} of the dataset j; the average accuracy (denoted by Avg_a_i) and the average precision (denoted by Avg_p_i) are defined as follows:

$$Avg\_a_i = \frac{\sum_{j=1}^{D} a_{i,j}}{D} \qquad Avg\_p_i = \frac{\sum_{j=1}^{D} p_{i,j}}{D}$$

where D denotes the number of datasets.

5.3 Related ranking approaches for comparison purposes

There are two other works on scientific paper ranking:

1. PTRA (Hasson et al., 2014),
2. ID3 (Rúbio & Gulo, 2016).

PTRA and ID3 are described in section 2.1. Table A 3.7 presents a summary of the criteria taken into account by each ranking approach: the bottom line of Table A 3.7 lists all the criteria used in the STELLAR ranking approach.

Table A 3.7 Criteria taken into account in three paper ranking approaches

Approaches	Year of publication	Citation number	Reference	Venue type	Venue age	Authors' impact	Citation category	Venue impact	Authors' institutes	Citing document of cited document
PTRA (Hasson et al., 2014)	X	X		X						
ID3 (Rúbio & Gulo, 2016)	X	X	X	X						
STELLAR	X	X	X	X	X	X	X	X	X	X

The performance of the STELLAR approach was compared against the performance of PTRA (Hasson et al., 2014) and ID3 (Rúbio & Gulo, 2016) on the same datasets and scenarios. In Table A 3.7, it is observed that for ranking a cited document as relevant, STELLAR considers more criteria, such as venue age, citation category, authors' impact, etc.

5.4 Analysis of the simulation results

This section presents the analysis of the simulation results in terms of papers' relevancy for the two datasets.

5.4.1 Simulation using the dataset harvested from databases

Figure A 3.17 shows the average accuracy for the three different simulations (STELLAR, ID3 and PTRA). The horizontal axis represents the sequence number of the simulation scenarios and the vertical axis represents the average accuracy of the associated scenario.

It is observed that STELLAR (in red) performs better than ID3 (in green) and PTRA (in blue): STELLAR has an average accuracy of 0.91 per scenario while ID3 has an average of 0.60 per scenario. The average relative improvement in accuracy (defined as [Avg_a of STELLAR – Avg_a of ID3]) of STELLAR in comparison to ID3 is 0.32 (32%) per scenario.

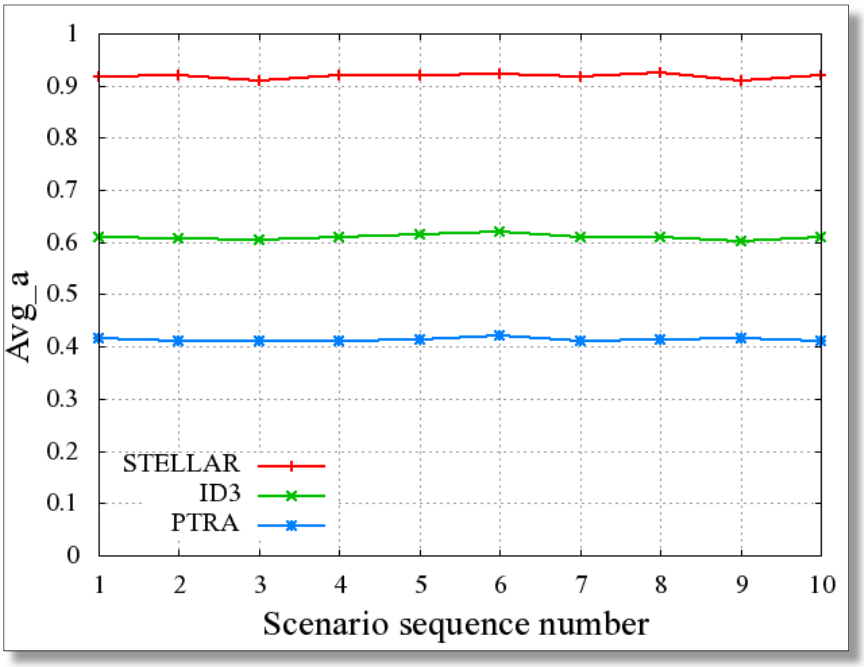


Figure A 3.17 Average accuracy vs Scenario sequence number – Harvested from databases

Figure A 3.18 shows the average precision for the same scenarios of Figure A 3.17. The x-axis represents the simulations scenario sequence number while the y-axis represents the average precision of the associated scenario.

STELLAR performed better than ID3 and PTRA: it produced an average precision of 0.96 per scenario while ID3, the better of the two approaches used for comparison, had an average of 0.65 per scenario. The average relative improvement in precision (defined as [Avg_p of STELLAR – Avg_p of ID3]) of STELLAR in comparison to ID3 is 0.31 (31%) per scenario.

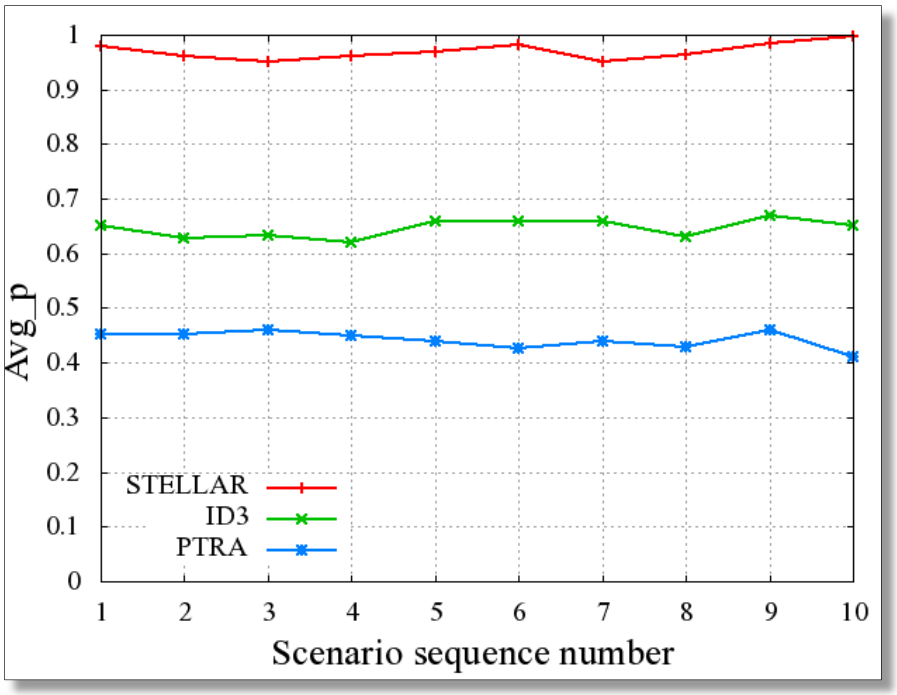


Figure A 3.18 Average precision vs Scenario sequence number – Harvested from databases

In both simulations and criteria, STELLAR outperformed ID3 and PTRA. This superior performance might be attributable to the use of additional bibliometric metadata to evaluate the relevancy of papers.

5.4.2 Simulation using the baseline dataset

Table A 3.8 presents the accuracy and precision when the list of papers in the baseline dataset (i.e., the references cited in this paper) is used as the dataset for simulations with the three ranking approaches.

Table A 3.8 Summary of performance criteria (accuracy and precision) using the baseline dataset

Approaches	Avg_a (%)	Avg_p (%)
PTRA (Hasson et al., 2014)	39.19	27.16
ID3 (Rúbio & Gulo, 2016)	53.98	41.97
STELLAR	76.09	68.73

1. STELLAR produced an average accuracy (Avg_a) of 76.09% while ID3 produced an accuracy of 53.98%. The relative improvement in accuracy of STELLAR as compared to ID3 is 22.11%.
2. STELLAR produced an average precision (Avg_p) of 68.73% while ID3 produced a precision of 41.97%. The relative improvement in precision of STELLAR as compared to ID3 is 26.76%.

Note that all the simulations are based on limited datasets, and should be extended later to larger datasets.

5.5 STELLAR prototype

This section presents a number of STELLAR's input screens. For example, Figure A.19 shows the input screen that allows researchers to enter their selections (RS) parameters.

STELLAR

Evaluated-based selection parameters:

Title	Latest developments in Semantic Web technologies applied to the glycosciences
Main topic	semantic web
Keywords	Semantic Web; Glycan repository; Glycan text representation
Description	important in and of themselves, their integration with other—omics data such as the protein information in UniProt will be crucial to bring glycosciences to the forefront of life sciences. However, to integrate such disparate sets of data among different fields in a way such that future maintenance costs are minimal, standardized ontologies and formats must be established. Our latest project has attempted to define the minimal standards that are necessary to enable this integration. The technical challenges to integrate all these databases and the technologies to overcome these challenges will be described.

Sort-based selection parameters:

LCR threshold	Enter radius threshold
Number of references	Enter number of references
MLTC percentage	Enter percentage of MLTC
MLTC year	Enter year of MLTC

Selected-based selection parameters:

Discipline	Computer Science
Language	English

Figure A 3.19 STELLAR input screen for researcher selection (RS) parameters

Figure A 3.20 shows a list of papers according to the RS parameters and their Literature Corpus radius (LCR). The paper's title is in the left column and its LCR is in the right column. Note that this list is ordered according to ascending LCR: the papers at the top are those that are closer to the RS parameters.

latest developments in semantic web technologies applied to the glycosciences	0.00
sustainable building technology knowledge representation: using semantic web techniques	44.00
biological data integration using semantic web technologies	44.00
tracing known security vulnerabilities in software repositories – a semantic web enabled modeling approach	45.00
semantic web technologies for supporting learning assessment	45.00
combining semantic web technologies with multi-agent systems for integrated access to biological resources	45.00
tap: a semantic web platform	45.00
context-awareness in the software domain—a semantic web enabled modeling approach	46.00
an application of intelligent techniques and semantic web technologies in e-learning environments	46.00
knowledge editing and maintenance tools for a semantic portal in oncology	46.00

Figure A 3.20 List of papers according to LCR based on researcher selection (RS) parameters

It can be seen that the radius of the paper at the top of the list is 0.0: indeed, this is the target paper.

The rest of this section presents four specific ALR assistance tools, shown in the following diagrams:

1. Timeline of a Document-based Literature Corpus Radius – Figure A 3.21,
2. Document-based Literature Corpus Radius – Figure A 3.22,
3. Timeline of an Author-based Literature Corpus Radius – Figure A 3.23,
4. Author-based Literature Corpus Radius – Figure A 3.24.

Figure A 3.21 represents the timeline of a Document-based Literature Corpus radius, with the horizontal axis indicating the year of publication (here, from 2011 to 2016).

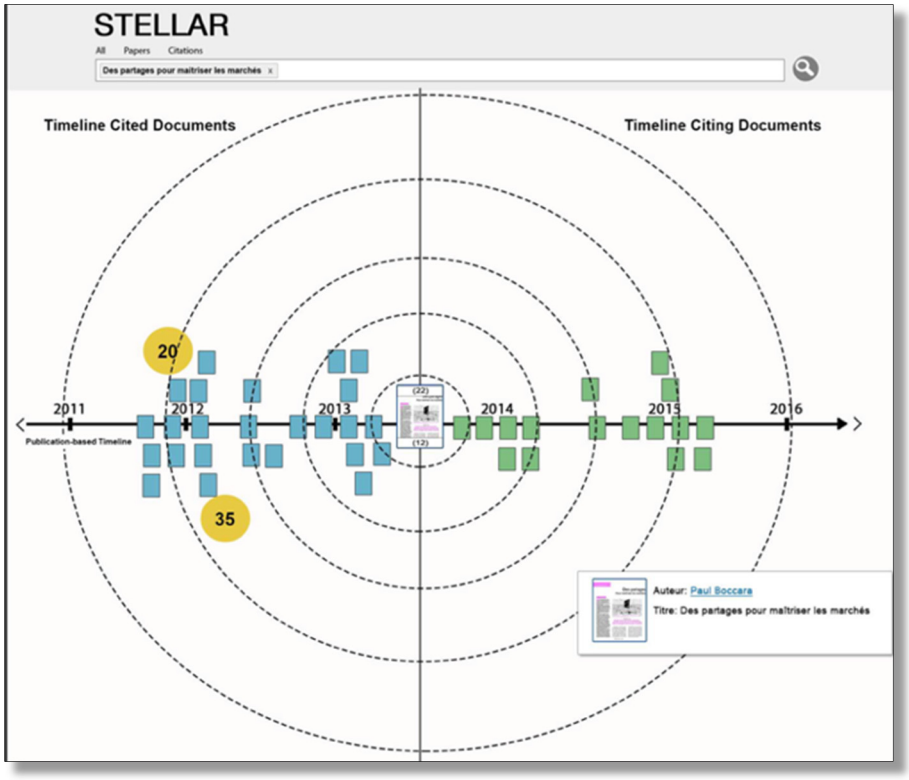


Figure A 3.21 Timeline of a Document-based Literature Corpus Radius (LCR)

In Figure A 3.21, the radius denotes the temporal distance from the document at center to the cited documents and to the citing documents. The yellow circles on the left side represent multiple documents—here, 20 to 35 documents.

Figure A 3.22 represents the Document-based Literature Corpus Radius model.

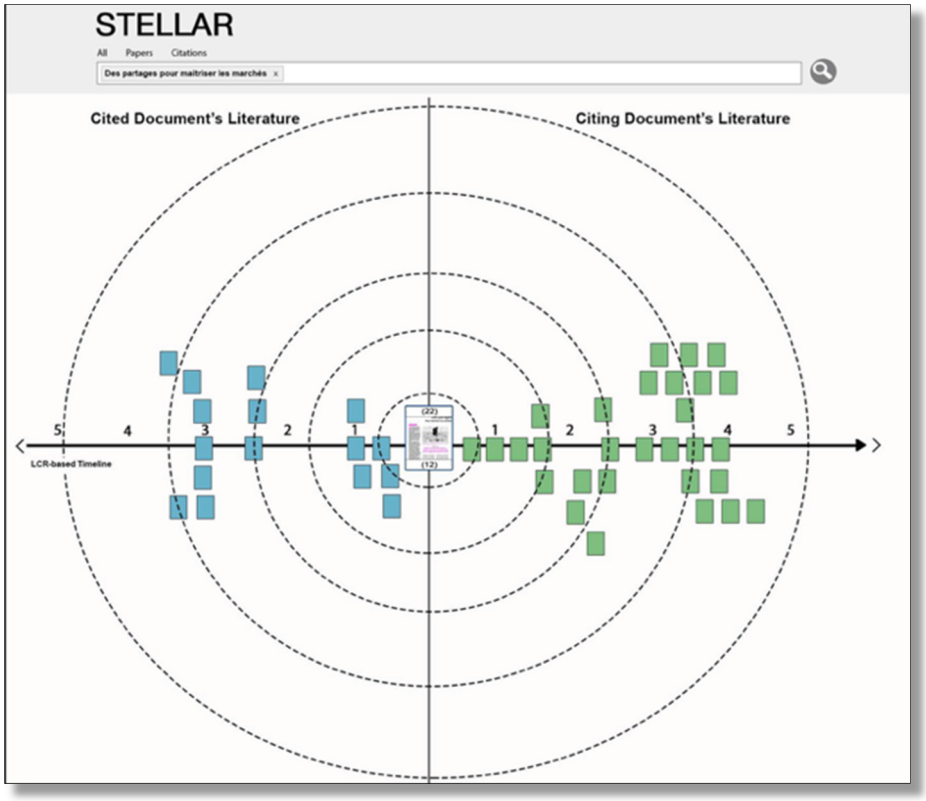


Figure A 3.22 Document-based Literature Corpus Radius (LCR)

The horizontal axis indicates the LCR: here, from 5 to 0 and from 0 to 5. The radius measures the distance from the center document to the cited document's literature (left side) and to its citing document's literature (right side).

The STELLAR prototype (Figure A 3.23) allows the researcher to view, for a given author (center document), the backward references (in blue) used and referred to by the document, as well as forward references (in green) to the center author (i.e., all documents referencing the center author).

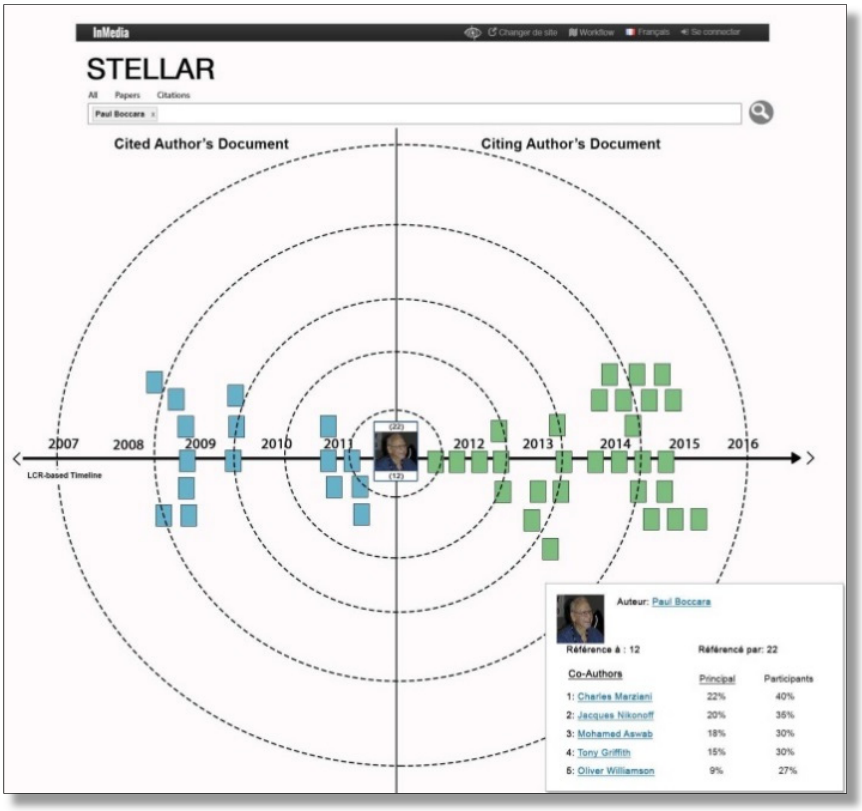


Figure A 3.23 Timeline of an Author-based Literature Corpus Radius - LCR

When any blue or green author is selected, the corresponding document will be re-positioned to the center, with all of its backwards references on the left in blue and all of its forward references (the ones citing the center author) on the right in green.

In this STELLAR prototype, the Author-based Literature Corpus Radius (Figure A 3.24) allows a researcher to view, for a given author (center author), the backward references (in blue) used and referred to by that author, and forward references (in green), i.e., all papers citing the center author.

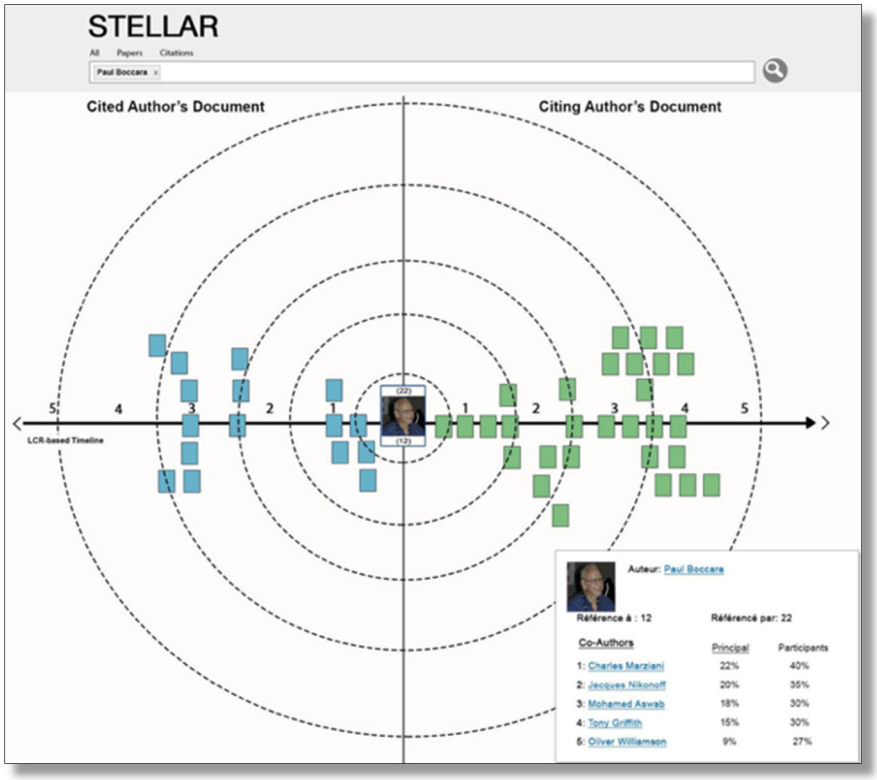


Figure A 3.24 Author-based Literature Corpus Radius (LCR)

6 Summary and Future Work

With the evolving, interdisciplinary nature of research and online access to research papers, there is a need to facilitate the iterative process of building a corpus for an assisted literature review (ALR). The aim of the present study is to assist researchers in finding, evaluating and annotating relevant papers, and to make them available at any time in an iterative process.

This paper has proposed an ALR prototype (STELLAR) based on machine learning model (MLM) and a semantic metadata ecosystem (SMESE) to identify, rank and recommend relevant papers for an ALR. Using text and data mining (TDM) models, MLM and a classification model that learns from researchers' annotated data (RA) and semantic enriched metadata, STELLAR assists in identifying and recommending papers that meet a researcher selection (RS) of parameters, including specific ALR topic, ALR title, ALR language, ALR discipline, ALR papers age, ALR number of references and other ALR metadata. The

STELLAR MLM produce an ALRO: they evaluate papers and related bibliographic attributes in order to determine their relevancy and ranking. Next, STELLAR aggregates all components related to the assisted creation of an ALR.

The STELLAR prototype presented in this paper is based on the Semantic Metadata Enrichment Software Ecosystem (SMESE V3), described in (Brisebois, Abran, Nadembega, et al., Unpublished results).

This paper has presented TDM models, related MLM and an enhanced metadata ecosystem that can help researchers produce ALRs. These include:

1. MLM designed to semantically harvest a Universal Research Documents Repository (URDR) according to a researcher selection and from the SMESE V3 ecosystem;
2. Literature Corpus Radius (LCR) MLM, which compute the distance from each paper to the center of the Literature Corpus defined by the researcher selection for a specific topic, concept or area of research;
3. MLM that help the researcher discover, find and refine the list of papers recommended for inclusion. To assist and narrow down the search results, many views of the ALR are made available to the researcher:
    a. Timeline of the Document-based Literature Corpus Radius,
    b. Document-based Literature Corpus Radius,
    c. Timeline of the Author-based Literature Corpus Radius,
    d. Author-based Literature Corpus Radius.

The performance of the STELLAR prototype has been evaluated through a comparison against a baseline manual LR using a number of simulations. In terms of accuracy, the STELLAR ALR provided an average accuracy of 0.91 per scenario while ID3 provided an average of 0.60 per scenario. In terms of precision, STELLAR produced an average of 0.96 per scenario while ID3 had an average of 0.65 per scenario. In comparison to ID3, STELLAR yielded an average relative improvement in accuracy of 32% per scenario and an average relative improvement in precision of 31%.

Figure A 3.25 presents the three areas of future work on the STELLAR prototype, the SMESE V3 platform (highlighted in blue boxes at the bottom right of Figure A 3.25) and Multi-Devices Content Machine Learning-based Assisted Recommendations:

1. Abstract of Abstracts summarization (AoA): AoA for scientific papers will be an extension of STELLAR; more specifically, abstracts will be used as input for our scientific paper summarization technique to generate the AoA.
2. Digital Resources Metadata Enrichment (DRME): the next STELLAR prototype will implement a new semantic discovery tool called DRME to help aggregate metadata from papers that have not published their metadata. DRME will use MLM to discover the metadata related to digital repositories and thus enrich digital resources.
3. Multi-Devices Content Machine Learning-based Assisted Recommendations. The purpose of this function will be to semantically match different types of content with the user's interests, availability and historical behavior, and to make suitable recommendations.

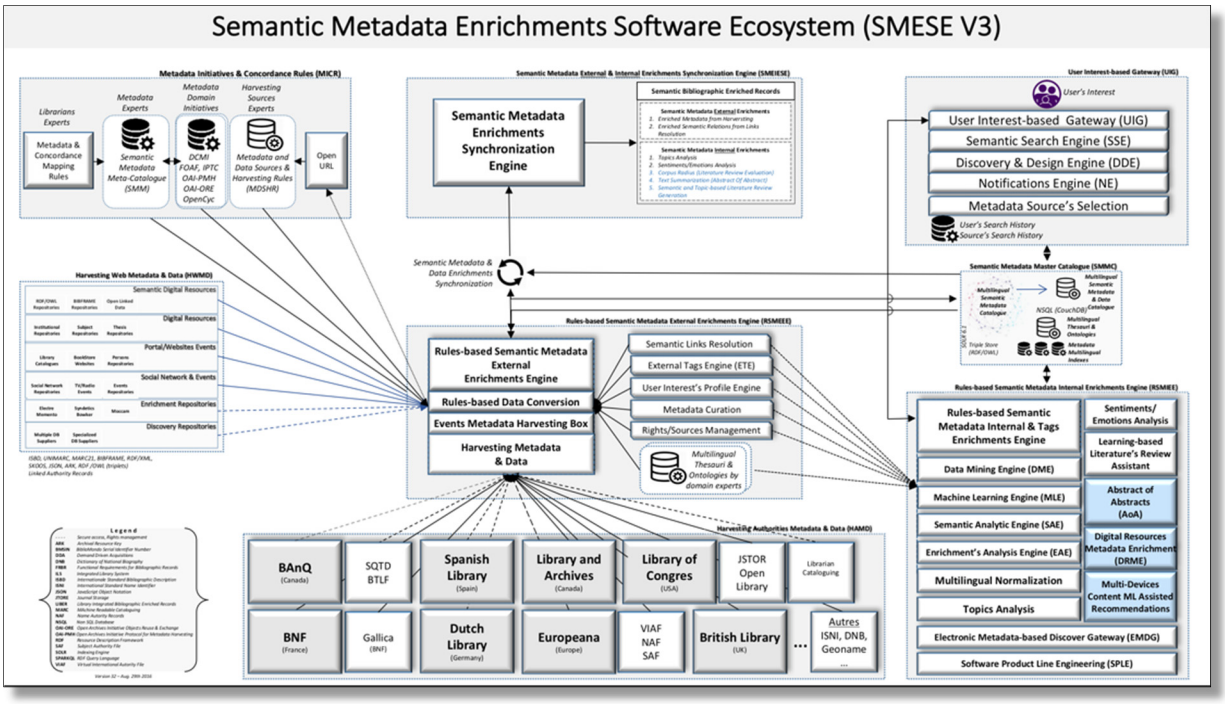


Figure A 3.25 Future contributions (in blue) to SMESE V3 platform

Furthermore, for a future version of STELLAR, we plan to work on MLM using learning process to enrich thesaurus as shown in Figure A 3.26.

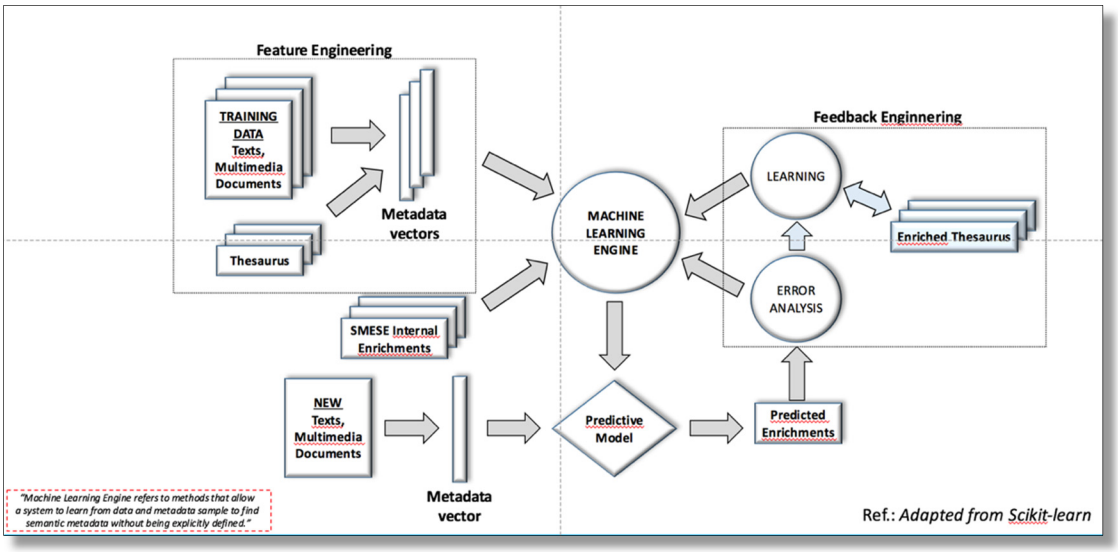


Figure A 3.26 STELLAR V2 future model

This STELLAR V2 will allow enhancing the SMESE V3 prototype to harvest semantic metadata from different sources as TV guides, radio channel schedule, books, music and other events calendar and create triplets to define relationships enriching metadata's content. A number of additional MLM, algorithms and prototypes will have to be developed and refined – see Figure A 3.27, including:

1. An algorithm to identify the Recommended User Interest-based New Content of Events (RUINCE criteria) representing the evolving interests and experience of users;
2. An algorithm to develop analytical recommendations of subscriptions about contents and events that will meet RUINCE criteria including the historical behaviour of the users;
3. An algorithm to recommend to user contents or events matching their interest or emotion according to the RUINCE affinity model;

4. An algorithm to rank dynamically the contents or events according to the RUINCE criteria to create interest-based channel's theme.

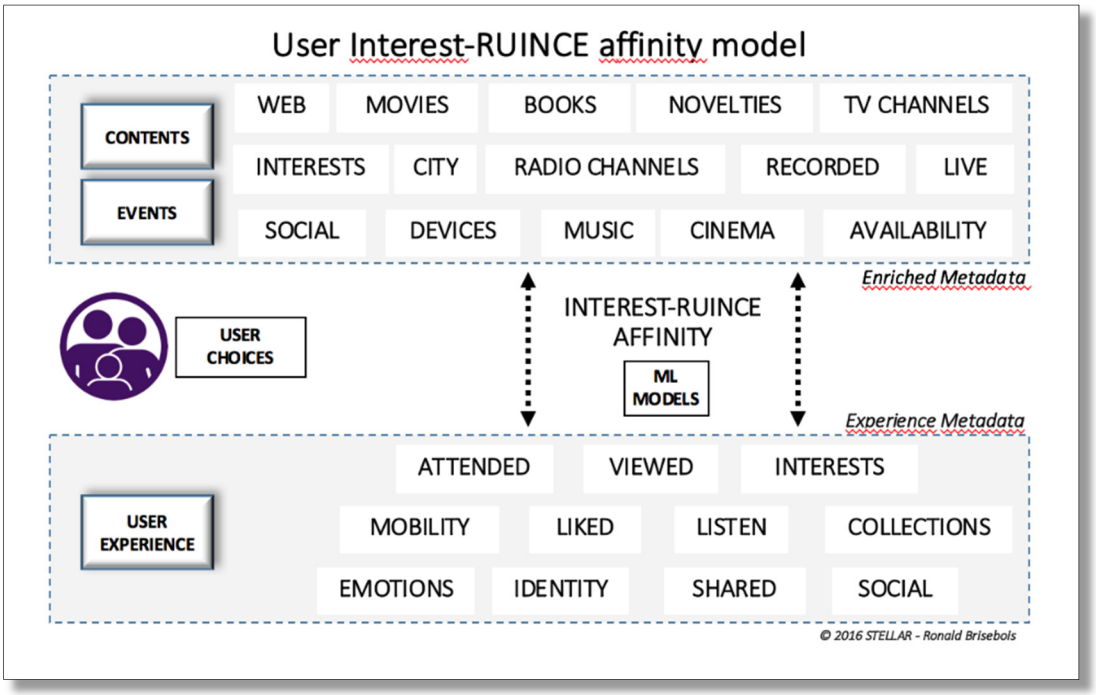


Figure A 3.27 User interest-RUINCE affinity metadata mapping model

Appendix A: Computation of the Literature Corpus Radius (LCR)

The literature corpus radius (LCR) is computed based on the evaluation-based parameters:

1. First, the value of each evaluation-based parameter is computed by determining the similarity of each evaluation-based selection with a predefined section of the document. The similarity matching value is in the range [0,1] where 1 means the most similar while 0 means the least similar.
2. Next, based on the similarity matching value (e.g., the predefined weight of each of them), the LCR index is computed.

- **Similarity matching of a researcher main topic with the topics extracted from documents abstracts**

The similarity matching with the researcher main topic is computed from the abstracts. The abstract of each document in the URDR is recorded in the "ABSTRACT" metadata provided by the publisher. The similarity matching computation makes use of this metadata as input to determine the document's similarity with the researcher-defined main topic.

Let d be the document and Ad the abstract of d. Next, based on the topic detection algorithm, called BM-Scalable Annotation-based Topic Detection (BM-SATD) (Brisebois, Abran, Nadembega, et al., Unpublished results), the topics of document d are detected from Ad. More specifically, BM-SATD uses multiple relations in a term graph and detects topics from the graph using a graph analytical method. Making use of document annotations, BM-SATD combines semantic relations between terms and co-occurrence relations across the document. Thus, using document abstracts as input, BM-SATD detects their topics.

Let:

1. *Ta* be the topic detected in the abstract of document *d*;
2. *MT* be the main topic provided as the researcher selection parameters and *n* be the number of terms of MT = (w_1, w_2, …, w_i, …, w_n);
3. *SimMatch_MaT(MT,d)* be the function that evaluates the similarity of *MT* with the document *d* abstract; note that the terms of *MT* are ordered.

First, the i-gram of MT is calculated in equation (A 3.1):

$$f(i-gram, MT, Ad) = \sum_{k=1}^{n-(i+1)} nb(w_k, w_{k+1}, \dots, w_{k+i-1}) \quad \text{(A 3.1)}$$

where nb(w_k, w_{k+1}, …, w_{k+i-1}) is the number of times that the i-gram (w_k, w_{k+1}, …, w_{k+i-1}) appear in *Ad* (the abstract of document *d*).

Next, the weight of the researcher's main topic for document *d* is computed using (A 3.2):

$$w\_MaT(MT,d) = \sum_{i=1}^{n} i \times f(i-gram, MT, Ad) \quad \text{(A 3.2)}$$

To obtain a similarity value between 0 and 1, normalization is applied. Let *Max_MaT* be the largest value of *w_MaT(MT,d)* among all the considered documents. SimMatch_MaT(MT,d) is computed using (A 3.3):

$$SimMatch\_MaT(MT,d) = \frac{w\_MaT(MT,d)}{Max\_MaT} \quad \text{(A 3.3)}$$

Thus, for each document, equations (A.1) to (A.3) compute the similarity of document with the researcher's main topic.

- **Similarity matching of researcher keywords with document keywords**

The similarity matching based on the researcher keywords is computed using the document keywords. The keywords of each document in the URDR are recorded in the "KEYWORDS" metadata provided by the publisher.

Let:

1. *Kd* be the set of keywords of document *d*;
2. *KW* be the set of keywords provided in the researcher selection parameters;
3. *SimMatch_KeW(KW,Kd)* be the function that computes the similarity matching of *KW* with *Kd*.

First, the weight of *KW* according to document *d* keywords *Kd* is computed as follows:

$$w\_KeW(KW,d) = |KW \cap Kd| \quad \text{(A 3.4)}$$

To obtain a similarity value between 0 and 1, normalization is applied; the SimMatch_KeW(KW,d) is computed as:

$$SimMatch\_KeW(KW,d) = \frac{w\_KeW(KW,d)}{|KW|} \quad \text{(A 3.5)}$$

Equations (A 3.4) to (A 3.5) compute the similarity of each document with the RS parameters in terms of keywords.

- **Similarity matching of researcher title with document titles**

Before the similarity matching computation, the researcher title and document titles are pre-processed. The objective of the pre-processing is to filter noise in order to obtain suitable text for performing the analysis. This consists in stemming, phrase extraction, part-of-speech filtering and removal of stop-words. More specifically, it includes the following operations:

1. Segmentation: the process of dividing a given document into sentences;
2. Stop-words removal: Stop-words are frequently occurring words (e.g., ‘a’ and ‘the’) that impart no meaning and generate noise. They are predefined and stored in an array. Note that the removal of stop-words follows specific rules. For example, in “prediction of mobility”, removal of the stop-word "of" changes the expression to "mobility prediction";
3. Tokenization: the input text is separated into tokens;
4. Punctuation marks: the spaces and word terminators are identified and treated as word breaking characters;
5. Word stemming: each word is converted into its root form by removing its prefix and suffix for comparison with other words.

The output of the pre-processing is the set of terms.

Let:

1. *Td* be the set of terms of the title of document *d*;
2. *TT* be the set of terms of the researcher selection title;
3. *SimMatch_TiT(TT,Td)* be the function that evaluates the similarity matching of TT with *Td*.

First, the weight of TT according to the document *d* title *Td* is computed as follows:

$$w\_TiT(TT,d) = \max_{j\in[1,m]}(j - gram(TT,Td)) \quad \text{(A 3.6)}$$

where m denotes the number of terms of TT (m=$|TT|$). Indeed, w_TiT(TT,d) is the largest number of sequential terms of TT that appears in Td. To obtain a similarity value between 0 and 1, normalization is applied. The SimMatch_TiT(TT,d) is computed as follows:

$$SimMatch\_TiT(TT,d) = \frac{w\_TiT(TT,d)}{m} \quad \text{(A 3.7)}$$

Thus, equations (A 3.6) to (A 3.7) compute the similarity matching of each document with the RS parameters "Title".

- **Similarity matching of the researcher description with document abstracts**

The similarity matching of the researcher research description is performed using the document abstract. To do this, the researcher description is semantically compared to the document abstract in order to measure the similarity level. This similarity matching of a researcher description makes use of WordNet::Similarity, described in (Pedersen et al., 2004), which implements six measures of similarity and three measures of relatedness. Several terms may be semantically the same.

Let:

1. *DS* be the researcher description of the research topic as the selection;
2. *s* be the number of terms of *DS* = (t_1, t_2, …, t_i, …, t_s);
3. *C* be the Literature Corpus where the documents are of the same discipline;
4. *SimMatch_DeC(DS,d)* be the function that evaluates the similarity matching of *DS* with a document abstract *Ad*.

First, the semantic similarity of each term in *DS* with those in *Ad* is determined on the basis of the semantic TF-ICF (term frequency – inverse corpus frequency) as follows:

$$SemSim\_T(t_i,d) = TF(t_i,d) \times \log\left(\frac{|C|}{ICF(t_i,C)}\right) \quad \text{(A 3.8)}$$

where $TF(t_i, d)$ and $ICF(t_i, d)$ denote the number of occurrences of t_i in document *d* and the number of documents in the corpus *C* where t_i appears.

Next, the semantic similarity of DS to the document abstract is computed as follows:

$$SemSim\_DeC(DS, d) = \sum_{i=1}^{s} SemSim\_T(t_i, d) \quad \text{(A 3.9)}$$

To obtain a similarity value between 0 and 1, normalization is applied. The SimMatch_DeC(DS,d) is computed as:

$$SimMatch\_DeC(DS, d) = \frac{SemSim\_DeC(DS, d)}{Max\_DeC} \quad \text{(A 3.10)}$$

where *Max_DeC* denotes the largest value of SemSim_DeC(DS,d) among all the documents in *C*.

Equations (A 3.8) to (A 3.10) compute the similarity matching of each document with the RS parameters "Description".

- **LCR index computation**

Once the similarity matching of each evaluation-based selection is done, the LCR index can be computed. An LCR index value is within the range [0,1] where 0 means the least similar while 1 is the most similar. Note that the LCR index is a weighted sum of the computed value of each selection.

Let:

1. W_init be an initial value,
2. W_unit be the difference in weight between two consecutive types of RS parameters.

The LCR index of a document *d* of literature corpus *C* is computed as follows:

$$Val(DS, d) = W\_init \times SimMatch\_DeC(DS, d) \quad \text{(A 3.11)}$$
$$Val(TT, d) = (W\_init + (W\_unit \times 1)) \times SimMatch\_TiT(TT, d)$$

$$Val(KW, d) = (W\_init + (W\_unit \ \times 2)) \times SimMatch\_KeW(KW, d)$$
$$Val(MT, d) = (W\_init + (W\_unit \ \times 3)) \times SimMatch\_MaT(MT, d)$$

$$LCR\ Index(d, MT, KW, TT, DS) =$$
$$1 - \left(\frac{Val(DS, d) + Val(TT, d) + Val(KW, d) + Val(MT, d)}{\sum_{i=0}^{3}(W\_init + (W\_unit \ \times i))}\right)$$

Appendix B: MLTC AND Number of references AND "To be included in the ALR" Pseudo-code

This appendix describes how STELLAR takes into account the researcher's requirements in terms of MLTC (Mix of the Literature Temporal Coverage (Yrs, %), number of references and the specific annotation "To be included in the ALR". The MLTC allows the researcher to include a certain percentage (%) of papers whose age is greater than a given age (Yrs). The idea here is to be able to include very relevant papers that are out of date. To take into account both the MLTC and the number of references without prioritizing either of them, a specific approach is needed, which is given by the following pseudo-code:

```
New_C1 = ∅
Old_C1 = ∅

  If (N ≤ Lengtj of All_C1)
          For the next  document in All_C1
             If [(A ≠ 0) AND (B ≠ 0)]
               If  [ (next  document publication age ≤ y) ]
                        Add next  document to New_C1
                        A=A-1
               Else If [ (next  document publication age >y) ]
                        Add next  document to Old_C1
```

```
                    B=B-1
            Else
        If [(A = 0) AND (B ≠ 0)]
                    Add next  document to Old_C1
                    B=B-1
                Else
                    If [(A ≠ 0) AND (B = 0)]
                        If  [ (next  document publication age ≤ y) ]
                            Add next  document to New_C1
                            A=A-1
    Else
            New_C1 = All_ C1
 C2= New_C1  ∪  Old_C1
```

Appendix C: ALR Index Categories

This appendix presents details on the three categories of indexes designed for the STELLAR prototypes:

1. Personal index,
2. Collaborative index,
3. DTb index.

a. Personal index

The DTb index identifies relevant documents in terms of scientific contributions in a specific domain and for a specific topic in order to generate an ALR.

However, the researcher may want only documents that he or she has tagged "To be included in the ALR". In this case, the personal index is computed in addition to the DTb index.

Let:

1. C_2 be the affinity match for ALR's LCR documents,
2. $d \in C_2$,
3. u be the researcher who requested the ALR.

The personal index is computed as follows:

$$Personal\ index(u,d) = \begin{cases} 1: \ document\ d\ is\ tagged\ by\ u \\ 0: \ document\ d\ is\ not\ tagged\ by\ u \end{cases} \quad \text{(A 3.12)}$$

Thus, for the personal index, all documents in C_2 whose personal index is 1 are selected.

b. Collaborative index

The collaborative index is also defined based on the documents that are tagged "To be included in the ALR" by a specific community of researchers or preselected researchers.

Let u_i be a researcher within the specific community of researchers or preselected researchers.

The collaborative index is computed as follows:

$$Collaborative\ index(u_i,d) = \begin{cases} 1: \ document\ d\ is\ tagged\ by\ u_i \\ 0: \ document\ d\ is\ not\ tagged\ by\ u_i \end{cases} \quad \text{(A 3.13)}$$

Thus, for the collaborative index, all the in C_2 whose collaborative index is 1 are selected.

c. Dynamic Topic based index

When a researcher does not clearly request a personal or collaborative index, a Dynamic Topic based index (DTb index) is applied to select documents relevant for the ALR. Like the LCR, the DTb index is also computed as a weighted sum of the values that denote the importance of the different inputs considered.

Note that paper topics are commonly used in the literature to compute the DTb index, and that publication dates and document ages are used regardless of their values. In STELLAR, therefore, the DTb index is computed using a number of additional concepts:

1. Key findings and peer citations index,
2. Venue index,
3. Document references index,
4. Authors and their affiliated institutes.

- **Document relevance according to researchers' key findings and peer citations**

The Key Findings are annotations in regards to important findings in the document related to the ALR. Indeed, previous researchers who have already analyzed these documents have

provided annotations called key findings. These key findings are identified and analyzed by the TDM approach. The TDM analysis consists in classifying the key findings into three categories:

1. *Very relevant:* indicates that the paper is very relevant and adequate for the LR,
2. *Adequate:* indicates that the paper is not relevant, but may be the focus of attention, if possible.
3. *Not relevant:* indicates that the paper is not relevant and not adequate for the search.

Let:

1. *Cat_annot* be the category of a key finding,
2. *Y* be the age of a document *d,*
3. *X* be the publication date of *d*.

For example: for a document published in 2000, Y =16 and X=2000.

The key findings index of document *d* is computed as follows:

$$KeyFindingsIndex(d, Cat\_Annot, Y) = \frac{\sum_{i=0}^{Y-1}[(Y-i) \times Nb(d, Cat\_Annot, (X+Y-i))]}{Y!} \quad \text{(A 3.14)}$$

where $\mathrm{Nb(d, Cat\_Annot, Z)}$ denotes the number of times the key findings Cat_Annot= "very relevant" are detected in document *d* at year Z.

The concept behind the computation of the key findings index is to give more importance to the more recent annotations instead of simply counting the number of considered key findings. This places more emphasis on recently published documents.

- **Document relevance according to venue**

The venue type is important in the ranking of scientific documents. The intent is to consider not only documents from academic journals, but also documents from other types of venues, such as conference proceedings and workshops, as well as unpublished documents such as research reports. In STELLAR, four types of venue are considered:

1. Journal,
2. Conference proceedings,
3. Workshop,
4. Unpublished.

Here, the venue types are ordered according to their importance in the researcher's opinion. For example: a researcher may consider that a journal paper is more important than a conference proceedings paper; thus, journal is first and conference is second. To compute the venue impact, the similarity matching of the detected topic with the venue main topic (where document *d* is published or presented) is computed as follows:

$$sim\_topic(Td, Tv) = \max_{j \in [1,m]} (j - gram(Td, Tv)) \quad \text{(A 3.15)}$$

where *Td* and *Tv* denote the detected topic of document *d* and the main topic of venue *v*, respectively.

The similarity matching between document title and venue name (where document *d* is published or presented) is computed as follows:

$$sim\_name(Nd, Nv) = \max_{j \in [1,m]} (j - gram(Nd, Nv)) \quad \text{(A 3.16)}$$

where *Nd* and *Nv* denote the title of document *d* and the name of venue *v*, respectively.

Thus, the venue *v* impact for a specific document d is given by:

$$\begin{aligned} VenueImpact(d, v) & \quad \text{(A 3.17)} \\ &= age\_venue(v) + avg\_num\_pub(v) \\ &+ rev\_num(v) + \frac{avg_{sub(v)}}{avg_{acc(v)}} + freq(v) \\ &+ sim\_topic(Td, Tv) + sim\_name(Nd, Nv) \end{aligned}$$

where

- *age_venue(v)* denotes the age of venue *v*,
- *avg_num_pu(v)* denotes the number of publications per year,

- *rev_num(v)* denotes the number of reviewers per submitted paper,
- *avg_sub(v)* denotes the average number of submitted papers per year,
- *avg_acc(v)* denotes the average number of accepted papers per year,
- *freq(v)* denotes the frequency of publication per year.

To take into account the type of venue, a weight is assigned to each of them according to its order and the couple (Vinit, Vunit), where:

- Vinit is an initial value and
- Vunit is the difference in weight between two consecutive types of venue.

For example: a venue type with order *i* will have the weight:

$$VtypeWeight(v) = Vinit + ((Q + 1 - i) \times Vunit\ ) \qquad \text{(A 3.18)}$$

where Q is the number of types of venue. Here, Q is equal to 4.

Finally, the venue-based index of document *d* is computed as follows:

$$VenueIndex(d, v) = VtypeWeight(v) \times VenueImpact(d, v) \qquad \text{(A 3.19)}$$

- **Document relevance according to authors and their affiliated institutes**

As was done for the venue index, the document relevance is computed on the basis of its authors and their affiliated institutes.

Let:

1. *Td* be the main topic of document *d*;
2. a_i be the author.

The influence on the document *d* is computed as follows:

$$AuthorImpact(d, a_i) = \frac{nb\_cited(Td)}{nb\_pub(Td)} + \frac{nb\_jour(Td)}{nb\_pub(Td)} + nb\_awar(Td, a_i) + nb\_jour(Td, I_i) + nb\_awar(Td, I_i) \quad \text{(A 3.20)}$$

where:

- *nb_cited(Td)* denotes the number of publications of author a_i cited on the topic *Td*,
- *nb_pub(Td)* denotes the number of publications of a_i on the topic *Td*,
- *nb_jour(Td)* denotes the number of journal publications by a_i on the topic *Td*,
- *nb_awar(Td,*a_i*)* denotes the number of awards of a_i on the topic *Td*,
- *nb_jour(Td,*I_i*)* denotes the number of publications which a_i's affiliated institute publishes in the most influential journals worldwide on the topic *Td*,
- *nb_awar(Td,*I_i*)* denotes the number of awards of a_i's affiliated institute on the topic *Td*.

The author index of document *d* is computed as follows:

$$AuthorsIndex(d) = \frac{\sum_{i=1}^{A}(A + 1 - i) \times AuthorImpact(d, a_i)}{A!} \quad \text{(A 3.21)}$$

where *A* denotes the number of authors of document *d*. The idea is to give more importance to top authors; the first author therefore has greater weight than the second author.

- **Document relevance according to document references**

The document's interaction with other documents on the topic is measured. Two groups of documents are defined:

1. Citing documents,
2. Cited documents.

For a better understanding, let *d* be a considered document; a citing document is a document that cited the document *d*, while a cited document is a document cited by the document *d*. Note

that the number of cited documents is static while the number of citing documents may increase with time. These two terms are important for the evaluation of document relevance. Figure A 3.14 illustrates the two terms according to the publication date.

The document's relevance based on citations includes several operands:

1. Number of citing documents according to the age of document *d*; it is computed as follows:

$$CitingImpact(d) = \frac{\sum_{i=0}^{Y-1}[(Y-i) \times nb\_citing(i+1)]}{Y!} \quad \text{(A 3.22)}$$

where *nb_citing(i)* denote the number of citing documents with age *i* and *Y* denotes the age of the document *d*. Relevant documents are those that are frequently cited. In addition, *CitingImpact (d)* gives more importance to recent citations.

2. Average number of times a document *d* is mentioned in citing documents; it is computed as follows:

$$CitingAvgImpact(d) = \frac{\sum_{j=1}^{P} nb\_time\_citing(d, D_j)}{P \times Y} \quad \text{(A 3.23)}$$

where $nb\_time\_citing(d,D_j)$, denotes the number of times the document *d* is cited in the citing document D_j, *P* is the total number of documents citing *d* and *Y* is the age of the document *d*.

$$CitedCitingAvgImpact(d) = \left| \bigcup_{D_l \in L} \left\{ \frac{nb\_citing(D_l)}{age(D_l)} \geq 5 \right\} \right| \quad \text{(A 3.24)}$$

where *L* denotes the set of documents cited in *d*, $age(D_l)$ denotes the age of document D_l and $nb\_citing(D_l)$ denotes the number of times document D_l is cited. Indeed, relevant documents very often cite existing relevant documents.

Finally, the relevancy of document *d* based on references is computed as follows:

$$ReferencesIndex(d) = CitingImpact(d) + CitingAvgImpact(d) + CitedCitingAvgImpact(d) \quad \text{(A 3.25)}$$

- **DTb index computation based on the previous computed index**

As mentioned above, the DTb index is a weighted sum of the computed values for different aspects that impact the relevancy of a document.

Let the couple (Init, Unit) where:

1. Init is an initial value, and
2. Unit is the difference in weight between two consecutive aspects.

The DTb index of document ***d*** is computed as follows:

$$Val(RF, d) = Init \times ReferencesIndex(d) \quad \text{(A 3.26)}$$

$$Val(VN, d) = (Init + (Unit \times 1)) \times VenueIndex(d, v)$$

$$Val(AA, d) = (Init + (Unit \times 2)) \times AuthorsIndex(d)$$

$$Val(KF, d) = (Init + (Unit \times 3)) \times KeyFindingsIndex(d, Cat\_Annot, Y)$$

$$DTb\ index(d, RF, VN, AA, KF) = \frac{Val(RF, d) + Val(VN, d) + Val(AA, d) + Val(KF, d)}{\sum_{k=0}^{3}(Init + (Unit \times k))}$$

LIST OF REFERENCES

Abdul-Mageed, M., Diab, M., & Kübler, S. (2014). SAMAR: Subjectivity and sentiment analysis for Arabic social media. *Computer Speech & Language, 28*(1), 20-37. doi:http://dx.doi.org/10.1016/j.csl.2013.03.001

Agarwal, N., Gvr, K., Reddy, R. S., & Rose, C. P. (2011). *SciSumm: a multi-document summarization system for scientific articles*. Paper presented at the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Systems Demonstrations, Portland, Oregon, USA.

Albert, B. E., Santos, R. P. d., & Werner, C. M. L. (2013, 24-26 July 2013). *Software ecosystems governance to enable IT architecture based on software asset management.* Paper presented at the 2013 7th IEEE International Conference on Digital Ecosystems and Technologies (DEST), Menlo Park, CA, USA.

Aleti, A., Buhnova, B., Grunske, L., Koziolek, A., & Meedeniya, I. (2013). Software Architecture Optimization Methods: A Systematic Literature Review. *IEEE Transactions on Software Engineering, 39*(5), 658-683. doi:10.1109/TSE.2012.64

Alférez, G. H., Pelechano, V., Mazo, R., Salinesi, C., & Diaz, D. (2014). Dynamic adaptation of service compositions with variability models. *Journal of Systems and Software, 91*, 24-47. doi:http://dx.doi.org/10.1016/j.jss.2013.06.034

Amorim, S. d. S., Almeida, E. S. D., & McGregor, J. D. (2013). *Extensibility in ecosystem architectures: an initial study*. Paper presented at the Proceedings of the 2013 International Workshop on Ecosystem Architectures, Saint Petersburg, Russia.

Andrés, C., Camacho, C., & Llana, L. (2013). A formal framework for software product lines. *Information and Software Technology, 55*(11), 1925-1947. doi:http://dx.doi.org/10.1016/j.infsof.2013.05.005

Andrzejewski, D., Zhu, X., & Craven, M. (2009). *Incorporating domain knowledge into topic modeling via Dirichlet Forest priors*. Paper presented at the Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, Quebec, Canada.

Anusha, V., & Sandhya, B. (2015). A Learning Based Emotion Classifier with Semantic Text Processing. In M. E.-S. El-Alfy, M. S. Thampi, H. Takagi, S. Piramuthu, & T. Hanne (Eds.), *Advances in Intelligent Informatics* (pp. 371-382). Cham, Switzerland: Springer International Publishing.

Appel, O., Chiclana, F., Carter, J., & Fujita, H. (2016). A hybrid approach to the sentiment analysis problem at the sentence level. *Knowledge-Based Systems, 108*, 110-124. doi:http://dx.doi.org/10.1016/j.knosys.2016.05.040

Ayala, I., Amor, M., Fuentes, L., & Troya, J. M. (2015). A Software Product Line Process to Develop Agents for the IoT. *Sensors, 15*(7), 15640-15660 doi:10.3390/s150715640

Balazs, J. A., & Velásquez, J. D. (2016). Opinion Mining and Information Fusion: A survey. *Information Fusion, 27*, 95-110. doi:http://dx.doi.org/10.1016/j.inffus.2015.06.002

Bao, S., Xu, S., Zhang, L., Yan, R., Su, Z., Han, D., & Yu, Y. (2012). Mining Social Emotions from Affective Text. *IEEE Transactions on Knowledge and Data Engineering, 24*(9), 1658-1670. doi:http://dx.doi.org/10.1109/TKDE.2011.188

Bechhofer, S., Buchan, I., De Roure, D., Missier, P., Ainsworth, J., Bhagat, J., . . . Goble, C. (2013). Why linked data is not enough for scientists. *Future Generation Computer Systems, 29*(2), 599-611. doi:http://dx.doi.org/10.1016/j.future.2011.08.004

Beel, J., Langer, S., Genzmehr, M., Gipp, B., Breitinger, C., & Nurnberger, A. (2013). *Research paper recommender system evaluation: a quantitative literature survey*. Paper presented at the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation, Hong Kong, China.

Bertin, M., Atanassova, I., Sugimoto, C. R., & Lariviere, V. (2016). The linguistic patterns and rhetorical structure of citation context: an approach using n-grams. *Scientometrics, 109*(3), 1417-1434. doi:10.1007/s11192-016-2134-8

Bijalwan, V., Kumar, V., Kumari, P., & Pascual, J. (2014). KNN based Machine Learning Approach for Text and Document Mining. *International Journal of Database Theory and Application, 7*(1), 61-70. doi:http://dx.doi.org/10.14257/ijdta.2014.7.1.06

Blei, D. M., & Lafferty, J. D. (2005). *Correlated Topic Models*. Paper presented at the Proceedings of the 19th Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research, 3*, 993–1022.

Bontcheva, K., Kieniewicz, J., Andrews, S., & Wallis, M. (2015). Semantic Enrichment and Search: A Case Study on Environmental Science Literature. *D-Lib Magazine, 21*(1-2), 1-18.

Boote, D. N., & Beile, P. (2005). Scholars Before Researchers: On the Centrality of the Dissertation Literature Review in Research Preparation. *Educational Researcher, 34*(6), 3-15. doi:http://dx.doi.org/10.3102/0013189x034006003

Bornmann, L., Stefaner, M., Anegón, F. d. M., & Mutz, R. (2014). Ranking and mapping of universities and research-focused institutions worldwide based on highly-cited papers: A visualisation of results from multi-level models. *Online Information Review, 38*(1), 43-58. doi:http://dx.doi.org/doi:10.1108/OIR-12-2012-0214

Bornmann, L., Stefaner, M., Anegón, F. d. M., & Mutz, R. (2015). Ranking and mappping of universities and research-focused institutions worldwide: The third release of excellencemapping.net. *COLLNET Journal of Scientometrics and Information Management, 9*(1), 65-72. doi:http://dx.doi.org/10.1080/09737766.2015.1027090

Bosco, C., Patti, V., & Bolioli, A. (2013). Developing corpora for sentiment analysis: The case of irony and senti-tut. *IEEE Intelligent Systems, 28*(2), 55-63.

Bougiatiotis, K., & Giannakopoulos, T. (2016). *Content Representation and Similarity of Movies based on Topic Extraction from Subtitles*. Paper presented at the Proceedings of the 9th Hellenic Conference on Artificial Intelligence, Thessaloniki, Greece.

Brisebois, R., Abran, A., & Nadembega, A. (2016). *A Semantic Metadata Enrichment Software Ecosystem (SMESE) based on a Multi-platform Metadata Model for Digital Libraries*. International Journal for Digital Libraries.

Brisebois, R., Abran, A., & Nadembega, A. (Unpublished results). *A Semantic Metadata Enrichment Software Ecosystem (SMESE) based on a Multi-platform Metadata Model for Digital Libraries*. International Journal for Digital Libraries.

Brisebois, R., Abran, A., Nadembega, A., & N'techobo, P. (Unpublished results). *A Semantic Metadata Enrichment Software Ecosystem based on Sentiment/Emotion Analysis Enrichment (SMESE V3)*. Information Systems.

Cambria, E., Gastaldo, P., Bisio, F., & Zunino, R. (2015). An ELM-based model for affective analogical reasoning. *Neurocomputing, 149, Part A*, 443-455. doi:http://dx.doi.org/10.1016/j.neucom.2014.01.064

Capilla, R., Bosch, J., Trinidad, P., Ruiz-Cortés, A., & Hinchey, M. (2014). An overview of Dynamic Software Product Line architectures and techniques: Observations from research and industry. *Journal of Systems and Software, 91*, 3-23. doi:http://dx.doi.org/10.1016/j.jss.2013.12.038

Capilla, R., Jansen, A., Tang, A., Avgeriou, P., & Babar, M. A. (2016). 10 years of software architecture knowledge management: Practice and future. *Journal of Systems and Software, 116*, 191-205. doi:http://dx.doi.org/10.1016/j.jss.2015.08.054

Caragea, C., Bulgarov, F., Godea, A., & Das Gollapalli, S. (2014). *Citation-Enhanced Keyphrase Extraction from Research Papers: A Supervised Approach*. Paper presented at the Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar.

Carenini, G., Cheung, J. C. K., & Pauls, A. (2013). MULTI-DOCUMENT SUMMARIZATION OF EVALUATIVE TEXT. *Computational Intelligence, 29*(4), 545-576. doi:http://dx.doi.org/10.1111/j.1467-8640.2012.00417.x

Carlos, A. S. J. G., & Thiago, R. P. M. R. (2015). *Text Mining Scientific Articles using the R Language*. Paper presented at the 10th Doctoral Symposium in Informatics Engineering, Porto, Portugal.

Cataldi, M., Di Caro, L., & Schifanella, C. (2016). *Ranking Researchers Through Collaboration Pattern Analysis*. Paper presented at the European Conference on Machine Learning and Knowledge Discovery in Databases, Riva del Garda, Italy. http://dx.doi.org/10.1007/978-3-319-46131-1_11

CELEBI, E., & DOKUN, O. (2015). Single-Document summarization using Latent Semantic Analysis. *International Journal of Scientific Research in Information Systems and Engineering (IJSRISE), 1*(2), 57-64.

Chen, J., & Zhuge, H. (2014). Summarization of scientific documents by detecting common facts in citations. *Future Generation Computer Systems, 32*, 246-252. doi:http://dx.doi.org/10.1016/j.future.2013.07.018

Chen, L., Qi, L., & Wang, F. (2012). Comparison of feature-level learning methods for mining online consumer reviews. *Expert Systems with Applications, 39*(10), 9588-9601. doi:http://dx.doi.org/10.1016/j.eswa.2012.02.158

Chen, P., Zhang, N. L., Liu, T., Poon, L. K. M., & Chen, Z. (2016). Latent Tree Models for Hierarchical Topic Detection. *arXiv preprint arXiv:1605.06650 [cs.CL]*, 1-44.

Cho, H., Kim, S., Lee, J., & Lee, J.-S. (2014). Data-driven integration of multiple sentiment dictionaries for lexicon-based sentiment classification of product reviews. *Knowledge-Based Systems, 71*, 61-71. doi:http://dx.doi.org/10.1016/j.knosys.2014.06.001

Christensen, H. B., Hansen, K. M., Kyng, M., & Manikas, K. (2014). Analysis and design of software ecosystem architectures – Towards the 4S telemedicine ecosystem. *Information and Software Technology, 56*(11), 1476-1492. doi:http://dx.doi.org/10.1016/j.infsof.2014.05.002

Cigarrán, J., Castellanos, Á., & García-Serrano, A. (2016). A step forward for Topic Detection in Twitter: An FCA-based approach. *Expert Systems with Applications, 57*, 21-36. doi:http://dx.doi.org/10.1016/j.eswa.2016.03.011

Clark, J. (2013). Text Mining and Scholarly Publishing. *Publishing Research Consortium*.

Conroy, J. M., & Davis, S. T. (2015). *Vector Space and Language Models for Scientific Document Summarization*. Paper presented at the Conference of the North American Chapter of the

Association for Computational Linguistics – Human Language Technologies, Denver, Colorado, USA. http://www.aclweb.org/anthology/W15-1525

Cotelo, J. M., Cruz, F. L., Enríquez, F., & Troyano, J. A. (2016). Tweet categorization by combining content and structural knowledge. *Information Fusion, 31*, 54-64. doi:http://dx.doi.org/10.1016/j.inffus.2016.01.002

Dang, Q., Gao, F., & Zhou, Y. (2016). Early detection method for emerging topics based on dynamic bayesian networks in micro-blogging networks. *Expert Systems with Applications, 57*, 285-295. doi:http://dx.doi.org/10.1016/j.eswa.2016.03.050

de Marneffe M-C, MacCartney B, & Manning CD. (2006). *Generating typed dependency parsers from phrase structure parses* Paper presented at the fifth international conference on language resources and evaluation, GENOA , ITALY

Demir, K. A. (2015). Multi-View Software Architecture Design: Case Study of a Mission-Critical Defense System. *Computer and Information Science, 8*(4), 12-31.

Desmet, B., & Hoste, V. (2013). Emotion detection in suicide notes. *Expert Systems with Applications, 40*(16), 6351-6358. doi:http://dx.doi.org/10.1016/j.eswa.2013.05.050

Di Ruscio, D., Paige, R. F., Pierantonio, A., Hutchinson, J., Whittle, J., & Rouncefield, M. (2014). Model-driven engineering practices in industry: Social, organizational and managerial factors that lead to success or failure. *Science of Computer Programming, 89*, 144-161. doi:http://dx.doi.org/10.1016/j.scico.2013.03.017

Dong, Y., Johnson, R. A., & Chawla, N. V. (2016). Can Scientific Impact Be Predicted? *IEEE Transactions on Big Data, 2*(1), 18-30. doi:http://dx.doi.org/10.1109/TBDATA.2016.2521657

dos Santos, R., P. , Esteves, M., S. , Freitas, G., & de Souza, J. (2014). Using Social Networks to Support Software Ecosystems Comprehension and Evolution. *Social Networking, 3*(2), 108-118. doi:10.4236/sn.2014.32014

Dumais, S. T. (2004). Latent semantic analysis. *Annual Review of Information Science and Technology, 38*(1), 188-230. doi:10.1002/aris.1440380105

Dunne, C., Shneiderman, B., Gove, R., Klavans, J., & Dorr, B. (2012). Rapid understanding of scientific paper collections: Integrating statistics, text analytics, and visualization. *Journal of the American Society for Information Science and Technology, 63*(12), 2351-2369. doi:http://dx.doi.org/10.1002/asi.22652

Dyas-Correia, S., & Alexopoulos, M. (2014). Text and Data Mining: Searching for Buried Treasures. *Serials Review, 40*(3), 210-216. doi:http://dx.doi.org/10.1080/00987913.2014.950041

Fang, H., Lu, W., Wu, F., Zhang, Y., Shang, X., Shao, J., & Zhuang, Y. (2015). Topic aspect-oriented summarization via group selection. *Neurocomputing, 149, Part C*, 1613-1619. doi:http://dx.doi.org/10.1016/j.neucom.2014.08.031

Fernández-Gavilanes, M., Álvarez-López, T., Juncal-Martínez, J., Costa-Montenegro, E., & Javier González-Castaño, F. (2016). Unsupervised method for sentiment analysis in online texts. *Expert Systems with Applications, 58*, 57-75. doi:http://dx.doi.org/10.1016/j.eswa.2016.03.031

Ferreira, R., de Souza Cabral, L., Lins, R. D., Pereira e Silva, G., Freitas, F., Cavalcanti, G. D. C., . . . Favaro, L. (2013). Assessing sentence scoring techniques for extractive text summarization. *Expert Systems with Applications, 40*(14), 5755-5764. doi:http://dx.doi.org/10.1016/j.eswa.2013.04.023

Fileto, R., Bogorny, V., May, C., & Klein, D. (2015). Semantic enrichment and analysis of movement data: probably it is just starting! *SIGSPATIAL Special, 7*(1), 11-18. doi:10.1145/2782759.2782763

Fileto, R., May, C., Renso, C., Pelekis, N., Klein, D., & Theodoridis, Y. (2015). The Baquara2 knowledge-based framework for semantic enrichment and analysis of movement data. *Data & Knowledge Engineering, 98*, 104-122. doi:http://dx.doi.org/10.1016/j.datak.2015.07.010

Franceschini, F., Maisano, D., & Mastrogiacomo, L. (2015). Influence of omitted citations on the bibliometric statistics of the major Manufacturing journals. *Scientometrics, 103*(3), 1083-1122. doi:http://dx.doi.org/10.1007/s11192-015-1583-9

Gall, M. D., Borg, W. R., & Gall, J. P. (1996). *Educational research: An introduction* (Vol. 6th ed.(1996). xxii 788 pp). White Plains, NY: England: Longman.

Gangemi, A. (2013). *A Comparison of Knowledge Extraction Tools for the Semantic Web*. Paper presented at the 10th European Semantic Web Conference (ESWC), Montpellier, France. http://dx.doi.org/10.1007/978-3-642-38288-8_24

Gawer, A., & Cusumano, M. A. (2014). Industry Platforms and Ecosystem Innovation. *Journal of Product Innovation Management, 31*(3), 417-433. doi:10.1111/jpim.12105

Genest, P.-E., & Lapalme, G. (2012). *Fully abstractive approach to guided summarization*. Paper presented at the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers - Volume 2, Jeju Island, Korea.

Gerani, S., Mehdad, Y., Carenini, G., Ng, R. T., & Neja, B. (2014). *Abstractive Summarization of Product Reviews Using Discourse Structure*. Paper presented at the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar.

Ghapanchi, A. H., Wohlin, C., & Aurum, A. (2014). Resources contributing to gaining competitive advantage for open source software projects: An application of resource-based theory. *International Journal of Project Management, 32*(1), 139-152. doi:http://dx.doi.org/10.1016/j.ijproman.2013.03.002

Ghiassi, M., Skinner, J., & Zimbra, D. (2013). Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network. *Expert Systems with Applications, 40*(16), 6266-6282. doi:http://dx.doi.org/10.1016/j.eswa.2013.05.057

Ginters, E., Schumann, M., Vishnyakov, A., & Orlov, S. (2015). Software Architecture and Detailed Design Evaluation. *Procedia Computer Science, 43*, 41-52. doi:http://dx.doi.org/10.1016/j.procs.2014.12.007

Gulo, C. A. S. J., Rubio, T. R. P. M., Tabassum, S., & Prado, S. G. D. (2015). Mining Scientific Articles Powered by Machine Learning Techniques. *OASIcs-OpenAccess Series in Informatics, 49*, 21-28. doi:http://dx.doi.org/10.4230/OASIcs.ICCSW.2015.21

Harman, M., Jia, Y., Krinke, J., Langdon, W. B., Petke, J., & Zhang, Y. (2014). *Search based software engineering for software product line engineering: a survey and directions for future work*. Paper presented at the Proceedings of the 18th International Software Product Line Conference - Volume 1, Florence, Italy.

Hasan, K. S., & Ng, V. (2014). *Automatic Keyphrase Extraction: A Survey of the State of the Art*. Paper presented at the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, Maryland, USA.

Hashimoto, T., Kuboyama, T., & Chakraborty, B. (2015). *Topic extraction from millions of tweets using singular value decomposition and feature selection*. Paper presented at the 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), Hong Kong, China.

Hasson, M. A., Lu, S. F., & Hassoon, B. A. (2014). Scientific Research Paper Ranking Algorithm PTRA: A Tradeoff between Time and Citation Network. *Applied Mechanics and Materials, 551*, 603-611. doi:http://dx.doi.org/10.4028/www.scientific.net/AMM.551.603

He, W., & Xu, L. D. (2014). Integration of Distributed Enterprise Applications: A Survey. *IEEE Transactions on Industrial Informatics, 10*(1), 35-42. doi:10.1109/TII.2012.2189221

He, Z., Chen, C., Bu, J., Wang, C., Zhang, L., Cai, D., & He, X. (2015). Unsupervised document summarization from data reconstruction perspective. *Neurocomputing, 157*, 356-366. doi:http://dx.doi.org/10.1016/j.neucom.2014.07.046

Henderson-Sellers, B., Gonzalez-Perez, C., McBride, T., & Low, G. (2014). An ontology for ISO software engineering standards: 1) Creating the infrastructure. *Computer Standards & Interfaces, 36*(3), 563-576. doi:http://dx.doi.org/10.1016/j.csi.2013.11.001

Horcas, J.-M., Pinto, M., & Fuentes, L. (2016). An automatic process for weaving functional quality attributes using a software product line approach. *Journal of Systems and Software, 112*, 78-95. doi:http://dx.doi.org/10.1016/j.jss.2015.11.005

Huang, S., & Wan, X. (2013). *AKMiner: Domain-Specific Knowledge Graph Mining from Academic Literatures*. Paper presented at the 14th International Conference on Web Information Systems Engineering (WISE), Nanjing, China. http://dx.doi.org/10.1007/978-3-642-41154-0_18

Hurtado, J. L., Agarwal, A., & Zhu, X. (2016). Topic discovery and future trend forecasting for texts. *Journal of Big Data, 3*(1), 1-21. doi:http://dx.doi.org/10.1186/s40537-016-0039-2

Jaidka, K., Khoo, C. S. G., & Na, J. C. (2010). *Imitating Human Literature Review Writing: An Approach to Multi-document Summarization*. Paper presented at the 12th International Conference on Asia-Pacific Digital Libraries (ICADL), Gold Coast, Australia. http://dx.doi.org/10.1007/978-3-642-13654-2_14

Jaidka, K., Khoo, C. S. G., & Na, J. C. (2013a). *Deconstructing human literature reviews–a framework for multi-document summarization*. Paper presented at the 14th European Workshop on Natural Language Generation, Sofia, Bulgaria.

Jaidka, K., Khoo, C. S. G., & Na, J. C. (2013b). Literature review writing: how information is selected and transformed. *Aslib Proceedings, 65*(3), 303-325. doi:http://dx.doi.org/doi:10.1108/00012531311330665

Jansen, S., & Bloemendal, E. (2013). Defining App Stores: The Role of Curated Marketplaces in Software Ecosystems. In G. Herzwurm & T. Margaria (Eds.), *Software Business. From Physical Products to Software Services and Solutions: 4th International Conference, ICSOB 2013, Potsdam, Germany, June 11-14, 2013. Proceedings* (pp. 195-206). Berlin, Heidelberg: Springer Berlin Heidelberg.

Jeremić, Z., Jovanović, J., & Gašević, D. (2013). Personal learning environments on the Social Semantic Web. *Semantic Web - Linked Data for science and education, 4*(1), 23-51. doi:10.3233/SW-2012-0058

Kedar, S. V., Bormane, D. S., Dhadwal, A., Alone, S., & Agarwal, R. (2015). *Automatic Emotion Recognition through Handwriting Analysis: A Review*. Paper presented at the 2015 International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India.

Khriyenko, O., & Nagy, M. (2011). *Semantic Web-driven Agent-based Ecosystem for Linked Data and Services.* Paper presented at the Third International Conferences on Advanced Service Computing, Rome, Italy

Kiritchenko, S., Zhu, X., & Mohammad, S. M. (2014). Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research, 50*(1), 723-762. doi:http://dx.doi.org/10.1613/jair.4272

Kontopoulos, E., Berberidis, C., Dergiades, T., & Bassiliades, N. (2013). Ontology-based sentiment analysis of twitter posts. *Expert Systems with Applications, 40*(10), 4065-4074. doi:http://dx.doi.org/10.1016/j.eswa.2013.01.001

Krishnan, S., Strasburg, C., Lutz, R. R., Goseva-Popstojanova, K., & Dorman, K. S. (2013). Predicting failure-proneness in an evolving software product line. *Information and Software Technology, 55*(8), 1479-1495. doi:http://dx.doi.org/10.1016/j.infsof.2012.11.008

Krueger, R., Thom, D., & Ertl, T. (2015). Semantic Enrichment of Movement Behavior with Foursquare-A Visual Analytics Approach. *IEEE Transactions on Visualization and Computer Graphics, 21*(8), 903-915. doi:10.1109/TVCG.2014.2371856

Kunze, C., & Hecht, R. (2015). Semantic enrichment of building data with volunteered geographic information to improve mappings of dwelling units and population. *Computers, Environment and Urban Systems, 53*, 4-18. doi:http://dx.doi.org/10.1016/j.compenvurbsys.2015.04.002

Lacasta, J., Nogueras-Iso, J., Falquet, G., Teller, J., & Zarazaga-Soria, F. J. (2013). Design and evaluation of a semantic enrichment process for bibliographic databases. *Data & Knowledge Engineering, 88*, 94-107. doi:http://dx.doi.org/10.1016/j.datak.2013.10.001

Lécué, F., Tallevi-Diotallevi, S., Hayes, J., Tucker, R., Bicer, V., Sbodio, M., & Tommasi, P. (2014). Smart traffic analytics in the semantic web with STAR-CITY: Scenarios, system and lessons learned in Dublin City. *Web Semantics: Science, Services and Agents on the World Wide Web, 27–28*, 26-33. doi:http://dx.doi.org/10.1016/j.websem.2014.07.002

Ledeneva, Y., García-Hernández, R. A., & Gelbukh, A. (2014). *Graph Ranking on Maximal Frequent Sequences for Single Extractive Text Summarization*. Paper presented at the 15th International Conference on Intelligent Text Processing and Computational Linguistics, Kathmandu, Nepal. http://dx.doi.org/10.1007/978-3-642-54903-8_39

Lei, J., Rao, Y., Li, Q., Quan, X., & Wenyin, L. (2014). Towards building a social emotion detection system for online news. *Future Generation Computer Systems, 37*, 438-448. doi:http://dx.doi.org/10.1016/j.future.2013.09.024

Lettner, D., Angerer, F., Prahofer, H., & Grunbacher, P. (2014). *A case study on software ecosystem characteristics in industrial automation software*. Paper presented at the Proceedings of the 2014 International Conference on Software and System Process, Nanjing, China.

Li, W., & Xu, H. (2014). Text-based emotion classification using emotion cause extraction. *Expert Systems with Applications, 41*(4, Part 2), 1742-1749. doi:http://dx.doi.org/10.1016/j.eswa.2013.08.073

Lim, S. L., Bentley, P. J., Kanakam, N., Ishikawa, F., & Honiden, S. (2015). Investigating Country Differences in Mobile App User Behavior and Challenges for Software Engineering. *IEEE Transactions on Software Engineering, 41*(1), 40-64. doi:10.1109/TSE.2014.2360674

Lin, C., He, Y., Everson, R., & Ruger, S. (2012). Weakly Supervised Joint Sentiment-Topic Detection from Text. *IEEE Transactions on Knowledge and Data Engineering, 24*(6), 1134-1145. doi:http://dx.doi.org/10.1109/TKDE.2011.48

Madani, F., & Weber, C. (2016). The evolution of patent mining: Applying bibliometrics analysis and keyword network analysis. *World Patent Information, 46*, 32-48. doi:http://dx.doi.org/10.1016/j.wpi.2016.05.008

Manikas, K., & Hansen, K. M. (2013). Software ecosystems – A systematic literature review. *Journal of Systems and Software, 86*(5), 1294-1306. doi:http://dx.doi.org/10.1016/j.jss.2012.12.026

Marx, W., & Bornmann, L. (2016). Change of perspective: bibliometrics from the point of view of cited references—a literature overview on approaches to the evaluation of cited references in bibliometrics. *Scientometrics, 109*(2), 1397-1415. doi:http://dx.doi.org/10.1007/s11192-016-2111-2

MASIC, I., & BEGIC, E. (2016). Evaluation of Scientific Journal Validity, It's Articles and Their Authors. *Stud Health Technol Inform., 226*, 9-14. doi:http://dx.doi.org/10.3233/978-161499-664-4-93-5

Mayr, P., Scharnhorst, A., Larsen, B., Schaer, P., & Mutschke, P. (2014). *Bibliometric-Enhanced Information Retrieval*. Paper presented at the 36th European Conference on IR Research (ECIR), Amsterdam, The Netherlands. http://dx.doi.org/10.1007/978-3-319-06028-6_99

Mendoza, M., Bonilla, S., Noguera, C., Cobos, C., & León, E. (2014). Extractive single-document summarization based on genetic operators and guided local search. *Expert Systems with Applications, 41*(9), 4158-4169. doi:http://dx.doi.org/10.1016/j.eswa.2013.12.042

Mens, T., Claes, M., Grosjean, P., & Serebrenik, A. (2014). Studying Evolving Software Ecosystems based on Ecological Models. In T. Mens, A. Serebrenik, & A. Cleve (Eds.), *Evolving Software Systems* (pp. 297-326). Berlin, Heidelberg: Springer Berlin Heidelberg.

Metzger, A., & Pohl, K. (2014). *Software product line engineering and variability management: achievements and challenges*. Paper presented at the Proceedings of the on Future of Software Engineering, Hyderabad, India.

Mitchell, T. M. (1997). *Machine Learning*. New York: McGraw-Hill.

Mohammad, S., Dorr, B., Egan, M., Hassan, A., Muthukrishan, P., Qazvinian, V., . . . Zajic, D. (2009). *Using citations to generate surveys of scientific paradigms*. Paper presented at the Human

Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Boulder, Colorado, USA.

Moraes, R., Valiati, J. F., & Gavião Neto, W. P. (2013). Document-level sentiment classification: An empirical comparison between SVM and ANN. *Expert Systems with Applications, 40*(2), 621-633. doi:http://dx.doi.org/10.1016/j.eswa.2012.07.059

Moreo, A., Romero, M., Castro, J. L., & Zurita, J. M. (2012). Lexicon-based Comments-oriented News Sentiment Analyzer system. *Expert Systems with Applications, 39*(10), 9166-9180. doi:http://dx.doi.org/10.1016/j.eswa.2012.02.057

Mück, T. R., & Fröhlich, A. A. (2014). A metaprogrammed C++ framework for hardware/software component integration and communication. *Journal of Systems Architecture, 60*(10), 816-827. doi:http://dx.doi.org/10.1016/j.sysarc.2014.09.002

Munezero, M. D., Montero, C. S., Sutinen, E., & Pajunen, J. (2014). Are They Different? Affect, Feeling, Emotion, Sentiment, and Opinion Detection in Text. *IEEE Transactions on Affective Computing, 5*(2), 101-111. doi:http://dx.doi.org/10.1109/TAFFC.2014.2317187

Musil, J., Musil, A., & Biffl, S. (2013). *Elements of software ecosystem early-stage design for collective intelligence systems*. Paper presented at the Proceedings of the 2013 International Workshop on Ecosystem Architectures, Saint Petersburg, Russia.

Nenkova, A., & McKeown, K. (2012). A Survey of Text Summarization Techniques. In C. C. Aggarwal & C. Zhai (Eds.), *Mining Text Data* (pp. 43-76). Boston, MA: Springer US.

Neves, A. R. d. M., Carvalho, Á. M. G., & Ralha, C. G. (2014). Agent-based architecture for context-aware and personalized event recommendation. *Expert Systems with Applications, 41*(2), 563-573. doi:http://dx.doi.org/10.1016/j.eswa.2013.07.081

Ngan, L. D., & Kanagasabai, R. (2013). Semantic Web service discovery: state-of-the-art and research challenges. *Personal and Ubiquitous Computing, 17*(8), 1741-1752. doi:10.1007/s00779-012-0609-z

Niu, T., Zhu, S., Pang, L., & El Saddik, A. (2016). *Sentiment Analysis on Multi-View Social Data*. Paper presented at the 22nd International Conference on MultiMedia Modeling (MMM), Miami, FL, USA. http://dx.doi.org/10.1007/978-3-319-27674-8_2

Okerson, A. (2013). *Text & Data Mining - A Librarian Overview*. Paper presented at the 79th IFLA World Library and Information Congress, Singapore, Malaysia.

Olyai, A., & Rezaei, R. (2015). Analysis and Comparison of Software Product Line Frameworks. *Journal of Software, 10*(8), 991-1001 doi:10.17706/jsw.10.8.991-1001

Oussalah, M., Bhat, F., Challis, K., & Schnier, T. (2013). A software architecture for Twitter collection, search and geolocation services. *Knowledge-Based Systems, 37*, 105-120. doi:http://dx.doi.org/10.1016/j.knosys.2012.07.017

Packalen, M., & Bhattacharya, J. (2015). Neophilia Ranking of Scientific Journals. *National Bureau of Economic Research Working Paper Series, 21579*. doi:http://dx.doi.org/10.3386/w21579

Park, J.-G., & Lee, J. (2014). Knowledge sharing in information systems development projects: Explicating the role of dependence and trust. *International Journal of Project Management, 32*(1), 153-165. doi:http://dx.doi.org/10.1016/j.ijproman.2013.02.004

Patel, G. A., & Madia, N. (2016). A Survey: Ontology Based Information Retrieval For Sentiment Analysis. *International Journal of Scientific Research in Science, Engineering and Technology, 2*(2), 460-465.

Patil, S. R., & Mahajan, S. M. (2012). *Scientific Research Paper Summarization On The Basis Of Research Relevant Term Identification*. Paper presented at the International Conference and workshop on Emerging Trends in Technology (ICWET), Mumbai, Maharashtra, India.

Pedersen, T., Patwardhan, S., & Michelizzi, J. (2004). *WordNet::Similarity: measuring the relatedness of concepts*. Paper presented at the Demonstration Papers at Human Language Technology conference/North American chapter of the Association for Computational Linguistics (HLT-NAACL), Boston, Massachusetts, USA.

Pedram, V. A., & Omid, S. S. (2015). Scientific Documents Clustering Based on Text Summarization. *International Journal of Electrical and Computer Engineering (IJECE), 5*(4), 782-787.

Poria, S., Cambria, E., Hussain, A., & Huang, G.-B. (2015). Towards an intelligent framework for multimodal affective data analysis. *Neural Networks, 63*, 104-116. doi:http://dx.doi.org/10.1016/j.neunet.2014.10.005

Porter, M. F. (1980). An algorithm for suffix stripping. *Program, 14*(3), 130-137. doi:doi:10.1108/eb046814

Premjith, P. S., John, A., & Wilscy, M. (2015). *Metaheuristic Optimization Using Sentence Level Semantics for Extractive Document Summarization*. Paper presented at the 3rd International Conference on Mining Intelligence and Knowledge Exploration (MIKE), Hyderabad, India. http://dx.doi.org/10.1007/978-3-319-26832-3_33

Quadri, A., & Abubakar, M. (2015). Software Quality Assurance in Component Based Software Development – A Survey Analysis. *I.J. of Computer and Communication System Engineering (IJCCSE), 2*(2), 305-315.

Quan, C., & Ren, F. (2014). Unsupervised product feature extraction for feature-oriented opinion determination. *Information Sciences, 272*, 16-28. doi:http://dx.doi.org/10.1016/j.ins.2014.02.063

Ravi, K., & Ravi, V. (2015). A survey on opinion mining and sentiment analysis: Tasks, approaches and applications. *Knowledge-Based Systems, 89*, 14-46. doi:http://dx.doi.org/10.1016/j.knosys.2015.06.015

Rettinger, A., Losch, U., Tresp, V., D'Amato, C., & Fanizzi, N. (2012). Mining the Semantic Web. *Data Min. Knowl. Discov., 24*(3), 613-662. doi:10.1007/s10618-012-0253-2

Robillard, M. P., & Walker, R. J. (2014). An Introduction to Recommendation Systems in Software Engineering. In P. M. Robillard, W. Maalej, J. R. Walker, & T. Zimmermann (Eds.), *Recommendation Systems in Software Engineering* (pp. 1-11). Berlin, Heidelberg: Springer Berlin Heidelberg.

Ronzano, F., & Saggion, H. (2016). *An Empirical Assessment of Citation Information in Scientific Summarization*. Paper presented at the 21st International Conference on Applications of Natural Language to Information Systems (NLDB), Salford, UK. http://dx.doi.org/10.1007/978-3-319-41754-7_30

Rúbio, T. R. P. M., & Gulo, C. A. S. J. (2016). *Enhancing Academic Literature Review through Relevance Recommendation*. Paper presented at the 11th Iberian Conference on Information Systems and Technologies, Gran Canaria, Canary Islands, Spain.

Saggion, H., & Poibeau, T. (2013). Automatic Text Summarization: Past, Present and Future. In T. Poibeau, H. Saggion, J. Piskorski, & R. Yangarber (Eds.), *Multi-source, Multilingual Information Extraction and Summarization* (pp. 3-21). Berlin, Heidelberg: Springer Berlin Heidelberg.

Salatino, A. A., & Motta, E. (2016). *Detection of Embryonic Research Topics by Analysing Semantic Topic Networks*. Paper presented at the Semantics, Analytics, Visualisation: Enhancing Scholarly Data, Montreal, Quebec, Canada. http://oro.open.ac.uk/id/eprint/45846

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management, 24*(5), 513-523. doi:http://dx.doi.org/10.1016/0306-4573(88)90021-0

Sankarasubramaniam, Y., Ramanathan, K., & Ghosh, S. (2014). Text summarization using Wikipedia. *Information Processing & Management, 50*(3), 443-461. doi:http://dx.doi.org/10.1016/j.ipm.2014.02.001

Sayyadi, H., & Raschid, L. (2013). A Graph Analytical Approach for Topic Detection. *ACM Transactions on Internet Technology, 13*(2), 1-23. doi:http://dx.doi.org/10.1145/2542214.2542215

Serrano-Guerrero, J., Olivas, J. A., Romero, F. P., & Herrera-Viedma, E. (2015). Sentiment analysis: A review and comparative analysis of web services. *Information Sciences, 311*, 18-38. doi:http://dx.doi.org/10.1016/j.ins.2015.03.040

Shinozaki, T., Yamamoto, Y., & Tsuruta, S. (2015). Context-based counselor agent for software development ecosystem. *Computing, 97*(1), 3-28. doi:10.1007/s00607-013-0352-y

Shivhare, S. N., & Khethawat, S. (2012). *Emotion Detection from Text*. Paper presented at the Second International Conference on Computer Science, Engineering and Applications (ICCSEA), Delhi, India. http://arxiv.org/abs/1205.4944

Singh, P. K., Sangwan, O. P., Singh, A. P., & Pratap, A. (2015). A Framework for Assessing the Software Reusability using Fuzzy Logic Approach for Aspect Oriented Software. *I.J. Information Technology and Computer Science, 7*(2), 12-20. doi:10.5815/ijitcs.2015.02.02

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Comput. Linguist., 37*(2), 267-307. doi:10.1162/COLI_a_00049

Tan, L. K.-W., Na, J.-C., Theng, Y.-L., & Chang, K. (2012). Phrase-Level Sentiment Polarity Classification Using Rule-Based Typed Dependencies and Additional Complex Phrases Consideration. *Journal of Computer Science and Technology, 27*(3), 650-666. doi:http://dx.doi.org/10.1007/s11390-012-1251-y

Trinidad, P. (2012). *Automating the Analysis of Stateful Feature Models.* (Ph.D Ph.D. dissertation), University of Seville, Spain.

Tsuruoka, Y., & Tsujii, J. i. (2005). *Bidirectional inference with the easiest-first strategy for tagging sequence data*. Paper presented at the Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, British Columbia, Canada.

Urli, S., Blay-Fornarino, M., Collet, P., Mosser, S., & Riveill, M. (2014, 27-29 Aug. 2014). *Managing a Software Ecosystem Using a Multiple Software Product Line: A Case Study on Digital Signage Systems.* Paper presented at the 40th EUROMICRO Conference on Software Engineering and Advanced Applications, Verona, Italy.

Vilares, D., Alonso, M. A., & GÓMez-RodrÍGuez, C. (2015). A syntactic approach for opinion mining on Spanish reviews. *Natural Language Engineering, 21*(1), 139-163. doi:http://dx.doi.org/10.1017/S1351324913000181

Wan, X., & Liu, F. (2014). WL-index: Leveraging citation mention number to quantify an individual's scientific impact. *Journal of the Association for Information Science and Technology, 65*(12), 2330-1643. doi:http://dx.doi.org/10.1002/asi.23151

Wang, D., Zhu, S., Li, T., & Gong, Y. (2013). Comparative Document Summarization via Discriminative Sentence Selection. *ACM Transaction on Knowledge Discovery Data, 7*(1), 1-18. doi:http://dx.doi.org/10.1145/2435209.2435211

Wang, S., Xie, S., Zhang, X., Li, Z., Yu, P. S., & Shu, X. (2014). *Future Influence Ranking of Scientific Literature*. Paper presented at the Society for Industrial and Applied Mathematics (SIAM) International Conference on Data Mining, Philadelphia, Pennsylvania, USA. http://epubs.siam.org/doi/abs/10.1137/1.9781611973440.86

Widyantoro, D. H., & Amin, I. (2014). *Citation sentence identification and classification for related work summarization*. Paper presented at the International Conference on Advanced Computer Science and Information Systems (ICACSIS), Jakarta, Indonesia.

Yadav, H. B., & Yadav, D. K. (2015). A fuzzy logic based approach for phase-wise software defects prediction using software metrics. *Information and Software Technology, 63*, 44-57. doi:http://dx.doi.org/10.1016/j.infsof.2015.03.001

Yang, C., Liang, P., & Avgeriou, P. (2016). A systematic mapping study on the combination of software architecture and agile development. *Journal of Systems and Software, 111*, 157-184. doi:http://dx.doi.org/10.1016/j.jss.2015.09.028

Yeloglu, O., Milios, E., & Zincir-Heywood, N. (2011). *Multi-document summarization of scientific corpora*. Paper presented at the ACM Symposium on Applied Computing (SAC), TaiChung, Taiwan.

Zajic, D., Dorr, B. J., Lin, J., & Schwartz, R. (2007). Multi-candidate reduction: Sentence compression as a tool for document summarization tasks. *Information Processing & Management, 43*(6), 1549-1570. doi:http://dx.doi.org/10.1016/j.ipm.2007.01.016

Zhang, C., Wang, H., Cao, L., Wang, W., & Xu, F. (2016). A hybrid term–term relations analysis approach for topic detection. *Knowledge-Based Systems, 93*, 109-120. doi:http://dx.doi.org/10.1016/j.knosys.2015.11.006

Zhang, M., Zhang, X., & Hu, Y. (2015). *Ranking of Collaborative Research Teams Based on Social Network Analysis and Bibliometrics*. Paper presented at the 12th International Conference on Cooperative Design, Visualization, and Engineering (CDVE), Mallorca, Spain. http://dx.doi.org/10.1007/978-3-319-24132-6_30

Zhang, R., Li, W., Liu, N., & Gao, D. (2016). Coherent narrative summarization with a cognitive model. *Computer Speech & Language, 35*, 134-160. doi:http://dx.doi.org/10.1016/j.csl.2015.07.004

THESIS PUBLISHED ARTICLES

Paper 1:

A Semantic Metadata Enrichment Software Ecosystem (SMESE) Based on a Multi-Platform Metadata Model for Digital Libraries

Ronald Brisebois, Alain Abran, Apollinaire Nadembega

https://doi.org/10.4236/jsea.2017.104022

Scientific Research Publishing

Journal of Software Engineering and Applications, 2017, 10, 370-405
http://www.scirp.org/journal/jsea
ISSN Online: 1945-3124
ISSN Print: 1945-3116

A Semantic Metadata Enrichment Software Ecosystem (SMESE) Based on a Multi-Platform Metadata Model for Digital Libraries

Ronald Brisebois[1], Alain Abran[1], Apollinaire Nadembega[2*]

[1]École de Technologie Supérieure, Université du Québec, Montréal, Canada
[2]University of Montreal, Montréal, Canada
Email: ronald.brisebois.1@ens.etsmtl.ca, alain.abran@etsmtl.ca, *apollinaire.nadembega@umontreal.ca

How to cite this paper: Brisebois, R., Abran, A. and Nadembega, A. (2017) A Semantic Metadata Enrichment Software Ecosystem (SMESE) Based on a Multi-Platform Metadata Model for Digital Libraries. *Journal of Software Engineering and Applications*, 10, 370-405.
https://doi.org/10.4236/jsea.2017.104022

Received: February 28, 2017
Accepted: April 27, 2017
Published: April 30, 2017

Abstract

Software industry has evolved to multi-product and multi-platform development based on a mix of proprietary and open source components. Such integration has occurred in software ecosystems through a software product line engineering (SPLE) process. However, metadata are underused in the SPLE and interoperability challenge. The proposed method is first, a semantic metadata enrichment software ecosystem (SMESE) to support multi-platform metadata driven applications, and second, based on mapping ontologies SMESE aggregates and enriches metadata to create a semantic master metadata catalogue (SMMC). The proposed SPLE process uses a component-based software development approach for integrating distributed content management enterprise applications, such as digital libraries. To perform interoperability between existing metadata models (such as Dublin Core, UNIMARC, MARC21, RDF/RDA and BIBFRAME), SMESE implements an ontology mapping model. SMESE consists of nine sub-systems: 1) Metadata initiatives & concordance rules; 2) Harvesting of web metadata & data; 3) Harvesting of authority metadata & data; 4) Rule-based semantic metadata external enrichment; 5) Rule-based semantic metadata internal enrichment; 6) Semantic metadata external & internal enrichment synchronization; 7) User interest-based gateway; 8) Semantic master catalogue. To conclude, this paper proposes a decision support process, called SPLE decision support process (SPLE-DSP) which is then used by SMESE to support dynamic reconfiguration. SPLE-DSP consists of a dynamic and optimized metadata-based reconfiguration model. SPLE-DSP takes into account runtime metadata-based variability functionalities, context-awareness and self-adaptation. It also presents the design and implementation of a working prototype of SMESE applied to a semantic digital library.

DOI: 10.4236/jsea.2017.104022 April 30, 2017

Keywords

Digital Library, Metadata Enrichment, Semantic Metadata Enrichment, Software Ecosystem, Software Product Line Engineering

1. Introduction

With more and more data available on the web, how users search and discover contents is of crucial importance. There is growing research on interaction paradigms investigating how users may benefit from the expressive power of semantic web standards.

The semantic web may be defined as the transformation of the worldwide web to a database of linked resources, where data may be widely reused and shared [1]. Web services can be enhanced by drawing on semantically aware data made available by a variety of providers. In addition, as information discovery needs to become more and more challenging, traditional keyword-based information retrieval methods are increasingly falling short in providing adequate support. This retrieval problem is compounded by the poor quality of the metadata content in some digital collections.

SECO [2]-[17] is defined as the interaction of a set of actors on top of a common technological platform providing a number of software solutions or services [2] [3]. In SECO, internal and external actors create and compose relevant solutions together with a community of domain experts and users to satisfy customer needs within specific market segments. This poses new challenges since the software systems providing the technical basis of a SECO are being evolved by various distributed development teams, communities and technologies.

There is growing agreement for the general characteristics of SECO, including a common technological platform enabling outside contributions, variability-enabled architectures, tool support for product derivation, as well as development processes and business models involving internal and external actors. At least ten SECO characteristics have been identified [18] that focus on technical processes for development and evolution, see Table 1.

Table 1. SECO characteristics [18].

1	Internal and external developers
2	Evaluative common technological platform
3	Controlled central part
4	Enable outside contributions and extensions
5	Variability-enabled architecture
6	Shared core assets
7	Automated and tool-supported product derivation
8	Outside contributions included in the main platform
9	Tools, frameworks and patterns
10	Distribution channel

Gawer and Cusumano [19] have analyzed a wide range of industry examples of SECO and identified two predominant types of platforms:

1. Internal platforms (company or product): defined as a set of assets organized in a common structure from which a company can efficiently develop and produce a stream of derivative products.
2. External platforms (industry): defined as products, services, or technologies that act as a foundation upon which external innovators, organized as an innovative business ecosystem, can develop their own complementary products, technologies, or services.

Indeed, the new generation of SECO must be an integration of multi-platforms (internal and external) that allows the interaction of a set of internal and external actors.

Concurrently modern software demands more and more adaptive features, many of which must be performed dynamically. In this context, a collaborative platform is important in order to coordinate collaborative and distributed environments for development of SECO platforms.

Furthermore, as the requirement of SECO to support adaptation capabilities of systems is increasing in importance [20] it is recommended such adaptive features be included within software product lines (SPL) [21] [22] [23] [24]. The SPL concept is appealing to organizations dealing with software development that aims to provide a comprehensive model for an organization building applications based on a common architecture and core assets [20] [21].

SPLs have been used successfully in industry for building families of systems of related products, maximizing reuse, and exploiting their variable and configurable options [22].

SPL development can be divided into three interrelated activities:

1. Core assets development: may include architecture, reusable software components, domain models, requirement statements, documentation, schedules, budgets, test plans, test cases, process descriptions, modeling diagrams, and other relevant items used for product development.
2. Product development: represents activities where products are physically developed from core assets, based on the production plan, in order to satisfy the requirements of the SPL [25].
3. Management: involves the essential processes carried out at technical and organizational levels to support the SPL process and ensures that the necessary resources are available and well coordinated.

To develop and implement SPL the literature proposes several SPL frameworks [23] using a variety of CBSD approaches [26] [27] [28]:

1. COPA (component-oriented platform architecting): an SPL framework that is component-oriented.
2. FAST (family-oriented abstraction, specification and translation): a software development process that divides the process of a product line into three sections: domain qualification, domain engineering and application engineering.
3. FORM (feature-oriented reuse method): a feature-oriented method that, by

analyzing the features of the domain, uses these features to provide the SPL architecture. FORM focuses on capturing commonalities and differences of applications in a domain in terms of features and uses the analysis results to develop domain architectures and components.

4. Kobra: a component-oriented approach based on the UML features that integrate the two paradigms into a semantic, unified approach to software development and evolution.
5. QADA (quality-driven architecture design and analysis): a product line architecture design method that provides traceability between the product quality and design time quality assessment.

Semantic web [29] [30] [31] [32] [33] linked data is the most important concept to support Semantic Metadata Enrichment (SME) in a SECO architecture [34]-[40].

Today, semantic web technologies, for example in digital libraries, offer a new level of flexibility, interoperability and a way to enhance peer communication and knowledge sharing by expanding the usefulness of the digital libraries that in the future will contain the majority of data. Indeed, a semantic web engine, based on semantic web technology, ensures more closely relevant results based on the ability to understand the definition and user-specific meaning of the word or term being searched for. Semantic search of semantic web engines are better able to understand the context in which the words are being used, resulting in relevant results with greater user satisfaction. Unfortunately, in the public domain there is a scarcity of search engines that follow a semantic-based approach to searching and browsing data [33]. Furthermore, the web is currently not contextually organized.

Thus, to enrich web data by transforming it into knowledge accessible by users, we propose a multi-platform architecture, referred to as SMESE, which uses a CBSD approach to integrate distributed content management enterprise applications, such as libraries and the Software Product Line Engineering (SPLE) approach.

Our SMESE architecture includes mobile first design (MFD) and semantic metadata enrichment (SME) engines that consist of metadata and meta-entity enrichment based on mapping ontologies and a semantic master metadata catalogue (SMMC).

More specifically, our SMESE implements a new decision support process in the context of SPLE, called the SPLE decision support process (SPLE-DSP), a meta entity model that represents all library materials and a meta metadata model. SPLE-DSP allows support for metadata-based reconfiguration. It consists of a dynamic and optimized metadata based reconfiguration model (DOMRM) where users select their preferences in the market place.

The major contributions of this paper are:

1. Definition of a software ecosystem model that configures the application production process including software aspects based on a proposed CBSD and metadata-based SPLE approach.

2. Definition and partial implementation of semantic metadata enrichment using SPLE and a semantic master metadata catalogue (SMMC) to create a universal metadata knowledge gateway (UMKG).
3. Design and implementation of a SMESE prototype for a semantic digital library (Libër).

This paper proposes a semantic metadata enrichment software ecosystem (SMESE) to support multi-platform metadata driven applications, such as a semantic digital library. Based on mapping ontologies SMESE also integrates and enriches data and metadata to create a semantic master metadata catalogue (SMMC).

The remainder of the paper is organized as follows. Section 2 is a literature review. Section 3 presents the multi-platform architecture of the proposed SMESE, and Section 4, the related nine sub-systems. Section 5 presents the prototype of a SMESE implementation in an industry context. Section 6 presents a summary and ideas for future work.

2. Literature Review

A software product line (SPL) [20]-[25] [41] [42] is a set of software intensive systems that share a common and managed set of features satisfying the specific needs of a particular market segment developed from a common set of core assets in a prescribed way [21] [23]. SPL engineering aims at: effective utilization of software assets, reducing the time required to deliver a product, improving quality, and decreasing the cost of software products.

The following sub-sections present the four research axes related to our research:

1. Software product line engineering (SPLE).
2. SECO architecture using component integration and component evolution.
3. SECO architecture and SPLE.
4. Semantic metadata enrichment (SME).

The related works section is at the intersection of SPLE, service-oriented computing, cloud computing, semantic metadata and adaptive systems.

2.1. Software Product Line Engineering (SPLE)

The development of software involves requirements analysis, design, construction, testing, configuration management, quality assurance and more, where stakeholders always look for high productivity, low cost and low maintenance. This has led to software product line engineering (SPLE) [24] as a comprehensive model that helps software providers to build applications for organizations/clients based on a common architecture and core assets. SPLE deals with the assembly of products from current core assets, commonly known as components, within a component-based architecture [43] [44], and involves the continuous growth of the core assets as production proceeds.

Note that the following related works are organized according to two axes: organizational and technical.

An overview of SPLE challenges is presented in [21] [22] [24]. Metzger and Pohl [21] suggest that the successful introduction of SPLE heavily depends on the implementation of adequate organizational structures and processes. They also identify three trends expected from SPLE research in the next decade:

1. Managing variability in non-product-line settings.
2. Leveraging instantaneous feedback from big data and cloud computing during SPLE.
3. Addressing the open world assumption in software product line settings.

A survey of works on search based software engineering (SBSE) for SPLE is presented in Harman *et al.* [22] [24].

Capilla *et al.* [24] provide an overview of the state of the art of dynamic software product line architectures and identify current techniques that attempt to tackle some of the many challenges of runtime variability mechanisms. They also provide an integrated view of the challenges and solutions that are necessary to support runtime variability mechanisms in SPLE models and software architectures. According to them, the limitations of today's SPLE models are related to their inability to change the structural variability at runtime, provide the dynamic selection of variants, or handle the activation and deactivation of system features dynamically and/or autonomously. SPLE is, therefore, the natural candidate within which to address these problems. Since it is impossible to predict all the expected variability in a product line, SPLE must be able to produce adaptable software where runtime variations can be managed in a controlled manner. Also, to ensure performance in systems that have strong real-time requirements, SPLE must be able to handle the necessary adaptations and current reconfiguration tasks after the original deployment due to the computational complexity during variants selection.

Olyai and Rezaei [23] describe the issues and challenges surrounding SPLs, introduce some SPLE ecosystems and compare them, based on the issues and challenges, with a view to how each ecosystem might be improved. The issues and challenges are presented in terms of administrative and organizational aspects and technical aspects. The administrative and organizational comparison criteria include strategic plans of the organization while the technical comparison criteria include requirements, design, implementation, test and maintenance. According to them, there is not a single approach that takes into account all these criteria together. Also, no single approach takes into account metadata for implementation and testing.

2.2. SECO Architecture Using Components Integration and Components Evolution

Software ecosystems (SECO) [2] [3] [4] [10] [19] [35] [39] consist of multiple software projects, often interrelated to each other by means of dependency relationships. When one project undergoes changes and issues a new release, this may or may not lead other projects to upgrade their dependencies. Unfortunately, the upgrade of a component may create a series of issues. In their systematic

literature review of SECO research, Manikas and Hansen [2] report that while research on SECO is increasing:

1. There is little consensus on what constitutes a SECO.
2. Few analytical models of SECO exist.
3. Little research is done in the context of real-world SECO.

They define a SECO as the interaction of a set of actors on top of a common technological platform that results in a number of software solutions or services where each actor is motivated by a set of interests or business models while connected to the rest of the actors. They also identify three main components of SECO architecture:

1. SECO software engineering: focuses on technical issues related directly or indirectly to the technological platform.
2. SECO business and management: focuses on the business, organizational and management aspects.
3. SECO relationships: represent the social aspect of the architecture since it is essential for SPLE actors to interact among themselves and with the platform.

2.3. SECO Architecture and SPLE

This section focuses on SECO architecture related to SPLE, beginning with an industry perspective.

Christensen *et al.* [3] define the concept of SECO architecture as a set of structures comprised of actors and software elements, the relationships among them, and their properties. They present the Danish telemedicine SECO in terms of this concept, and discuss challenges that are relevant in areas beyond telemedicine. They also discuss how software engineering practice is affected by describing the creation and evolution of a central SECO architecture, namely Net4Care, that serves as a reference architecture and learning vehicle for telemedicine and for the actors within a single software organization.

Demir [34] also proposes a software architecture that is strongly related to a defence system and limited to military personnel. Their multi-view SECO architecture design is described step by step. They begin by identifying the system context, requirements, constraints, and quality expectations, but do not describe the end products of the SECO architecture. They also introduce a novel architectural style, called "star-controller architectural style" [34] where synchronization and control of the flow of information are handled by controllers. However, a major drawback of this style is that failure of one controller disables all the subcomponents attached to that controller.

Neves *et al.* [40] propose an architectural solution based on ontology and the spreading algorithm that offers personalized and contextualized event recommendations in the university domain. They use an ontology to define the domain knowledge model and the spreading activation algorithm to learn user patterns through discovery of user interests. The main limitation of their architectural context-aware recommender system is that it is specific to university populations and does not present the actual model of the system that shows the interactions between the components and the data.

Alferez *et al.* [45] propose a framework that uses semantically rich variability models at runtime to support the dynamic adaptation of service compositions. They argue that should problematic events occur, functional pieces may be added, removed, replaced, split or merged from a service composition at runtime, hence delivering a new service composition configuration. Based on this argument, they propose that service compositions be abstracted as a set of features in a variability model. They define a feature as a logical unit of behavior specified by a set of functional and non-functional requirements. Thus, they propose adaptation policies that describe the dynamic adaptation of a service composition in terms of the activation or deactivation of features in the causally connected variability model. Unfortunately, this variability model is limited to activation and deactivation of services. Indeed, the model should allow adaptation of services or include a service interoperability protocol (SIP) rather than compositions only according to changes in the computing infrastructure.

In component based software development (CBSD), the fuzzy logic approach [27] [28] is largely used to select components. Singh *et al.* [27] explored the various measures such as separation of concerns (SoC), coupling, cohesion, and size measure that affect the reusability of aspect oriented software. The main drawback of their contribution is that the fuzzy logic rules are static. They do not propose a way to improve the rules based on developer satisfaction of the fuzzy inference system (FIS) output. In addition, their fuzzy inference system is limited to reusability of software.

2.4. Semantic Metadata Enrichment (SME)

Bontcheva *et al.* [46] investigate semantic metadata automatic enrichment and search methods. In particular, the benefits of enriching articles with knowledge from linked open data resources are investigated with a focus on the environmental science domain. They also propose a form-based semantic search interface to facilitate environmental science researchers in carrying out better semantic searches. Their proposed model is limited to linking terms with DBpedia URI and does not take into account the semantic meaning of terms in order to detect the best DBpedia URI.

Some authors focus their enrichment model on person mobility trace data [47] [48] [49] [50]. Krueger *et al.* [47] show how semantic insights can be gained by enriching trajectory data with place of interest (POI) information using social media services. They handle semantic uncertainties in time and space, which result from noisy, imprecise, and missing data, by introducing a POI decision model in combination with highly interactive visualizations. However, this model is limited to POI detection.

Kunze and Hecht [48] propose an approach to processing semantic information from user-generated OpenStreetMap (OSM) data that specifies non-residential use in residential buildings based on OSM attributes, so-called tags, which are used to define the extent of non-residential use.

Our conclusions from these related works are:

1. SPLE architecture needs to be flexible and meet administrative and organizational aspects such as the organization's strategic plans and marketing strategies, as well as technical aspects such as requirements, design, implementation, test and maintenance.
2. Researchers need to focus on real-world SECO.
3. Several proposed SECO models do not take into account autonomic mechanisms to guide the self-adaptation of service compositions according to changes in the computing infrastructure.
4. In CBSD fuzzy inference systems (FIS) have been employed to develop the components selection model, however, there is no FIS based model that proposes more than one software measure as FIS output.
5. There is no SECO architecture that takes into account several semantic enrichment aspects.
6. Current metadata and entity enrichment models are limited to only one domain for their semantic enrichment process and therefore do not involve several enriched metadata and entity models.
7. Current metadata and entity enrichment models only link terms and DBpedia URI.
8. Current metadata and entity enrichment models do not take into account person mobility trace data gathering and analysis in the enrichment process of metadata.

3. SMESE Multi-Platform Architecture

This section presents the proposed semantic enriched metadata software ecosystem (SMESE) architecture based on SPLE and CBSD approaches to support metadata and entity social and semantic enrichment for semantic digital libraries and based on an MFD approach for user interface design. Each component of the SMESE architecture is based on existing approaches (SPLE and CBSD) and an SME concept (proposed in this work) to generate, extract, discover and enrich metadata based on mapping ontologies and making use of contents and linked data analysis.

For the new generation of information and data management, metadata is a most efficient material for data aggregation. For example, it is easier to find a specific set of interests for users based on metadata such as content topics, or based on the sentiments expressed in a content. Furthermore, it is possible to increase user satisfaction by reducing the user interest gap. To make this feasible, all content needs to be enriched. In other words, specific metadata must be available including semantic topics, sentiments and abstracts. However, at the present time more than 85% of content does not have this metadata.

The SMESE multiplatform prototype includes an engine to aggregate multiple world catalogues from libraries, universities, Bbookstores, #tag collections, museums, and cities. The collection of pre-harvested and processed metadata and full text comprises the searchable content.

Central indexes typically include: full text and citations from publishers, full

text and metadata from open source collections, full text, abstracting, and indexing from aggregators and subscription databases, and different formats (such as MARC) from library catalogues, also called the base index, unified index, or foundation index.

The SMESE multiplatform framework must link bibliographic records and semantic metadata enrichments into a digital world library catalogue. SMESE must search and discover actual collections or novelties, including: works, books, DVDs, CDs, comics, games, pictures, videos peoples, legacy collections, organizations, rewards, TVs, radios, and museums.

The five levels of the semantic collaborative gateway are:

1. Meta Entity.
2. Entity.
3. Semantic metadata enrichment and creation.
4. Free sources of metadata and subscription-based metadata.
5. Content.

Figure 1 presents the entity matrix. The metadata are defined once and are related to each specific entity.

Semantic relationships between the contents, persons, organization and places are defined and curated in the master metadata catalogue. Topics, sentiments and emotions must be extracted automatically from the contents and their context:

ENTITY (NOTICES) MATRIX of the SMESE's Master Catalogue (EXAMPLE)

Calendars
- *Interests*
- *Library*
- *POI*
- *Rewards*
- *TV Channel*
- ...

Collections
- *Interests*
- *Library*
- *Organizations*
- *Personal*
- ...

Contents
- *Audio Books*
- *Books*
- *Cartographic Mat.*
- *Citations*
- *Comics*
- *Estampes*
- *Manga*
- *Microforms*
- *Movies (DVD)*
- *Music (CD)*
- *Musical Partitions*
- *Old Books*
- *Photos (Image)*
- *Press*
- *Serials*
- *Sounds*
- *Videos*
- ...

Documents
- *Google Doc*
- *Paint*
- *PDF*
- *Powerpoint*
- *Spreadsheet*
- *Word*
- ...

Events
- *Cinemas Rep.*
- *Expotitions*
- *Libēr Spirits*
- *News*
- *Notifications*
- *Press Conference*
- *Shows*
- *Spectacles*
- *Theaters*
- *TV Shows*
- ...

- *ALRO*
- *Titre*
- *Topics*
- *Keywords*
- *#references*
- *Annotations*
- ...

Objects
- *Object*
- *Work of Art*
- ...

Persons
- *Actor*
- *Author*
- *Celebrity*
- *Musician*
- *Politician*
- *Producer*
- *Singer*
- *Students*
- *User*
- ...

Places
- *City*
- *Localization*
- *POI*
- ...

Products
- *Financial*
- *Groceries*
- *Hardware*
- *Natural Health*
- *Pharmacy*
- *Software*
- ...

Publications
- *Articles*
- *Education Programs*
- *Fact Sheets*
- *Questions/Answers*
- *Manuals*
- *Monographs*
- *Newsletters*
- *PostCards*
- *Posters*
- *Proceedings*
- *Thesis*
- ...

Rewards
- *Litterature*
- *Movies*
- *Music*
- *Nobel*
- ...

Resources
- *Online*
- *Physical*
- ...

Subjects
- *Genomes*
- *MindMaps*
- *Ontologies*
- ...

WebSites
- *Homework Help*
- *Youth*
- ...

Works
- *Concepts*
- *Expressions*
- *Manifestations*
- ...

Figure 1. Entity matrix.

1. Libraries spend a lot of money buying books and electronic resources. Enrichment uncovers that information and makes it possible for people to discover the great resources available everywhere.
2. The average library has hundreds of thousands of catalogue records waiting to be transformed into linked data, turning those thousands of records into millions of relationships.

FRBR (functional requirements for bibliographic records) is a semantic representation of the bibliographic record. A work is a high-level description of a document, containing information such as author (person), title, descriptions, subjects, etc., common to all expressions, format and copy of the work (see Figure 2 for an FRBR framework description).

SMESE must allow users to find topically related content through an interest-based search and discovery engine. Transforming bibliographic records into semantic data is a complex problem that includes interpreting and transforming the information. Fortunately, many international organizations (e.g., BNF, Library of Congress and some others) have partly done this heavy work and already have much bibliographic metadata converted into triple-stores.

Recent catalogues support the ability to publish and search collections of descriptive entities (described by a list of generic metadata) for data, content, and related information objects. Metadata in catalogues represent resource characteristics that can be indexed, queried and displayed by both humans and software. Catalogue metadata are required to support the discovery and notification of information within an information community. Using the information from these Semantic Metadata Enrichments, the search engine, discovery engine and notification engine are able to give to the final user better results in accord with his interest or mood.

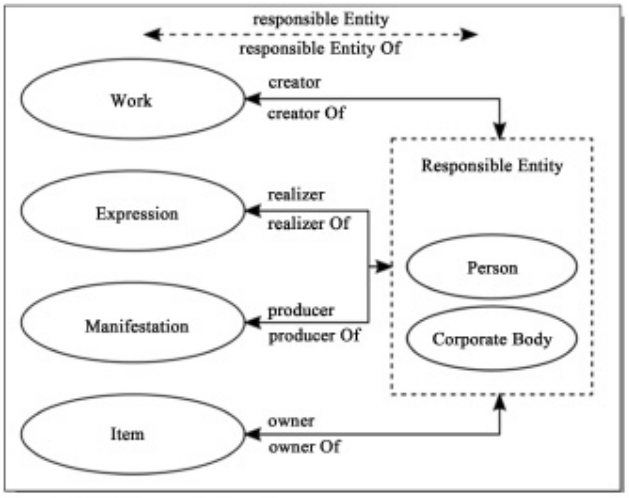


Figure 2. FRBR framework description.

SMESE must also include an automated approach for semantic metadata enrichment (SME) that allows users to perform interest-based semantic search or discovery more efficiently. To summarize, our SMESE makes the following contributions:

Definition and development of a proposed semantic metadata enrichment software ecosystem (see Figure 3 for SMESE overview and Appendix B shows the detailed version).

This new semantic ecosystem will harvest and enrich bibliographic records externally (from the web) and internally (from text data). The main components of the ecosystem will be:

1. Metadata initiatives & concordance rules
2. Harvesting web metadata & data
3. Harvesting authority metadata & data
4. Rule-based semantic metadata external enrichment engine
5. Rule-based semantic metadata internal enrichment engine
6. Semantic metadata external & internal enrichment synchronization engine
7. User interest-based gateway
8. Semantic master catalogue

A. *Topic detection/generation*: A prototype was developed to automate the generation of topics from the text of a document using our algorithm BM-SATD (Semantic Annotation-based Topic Detection). In this research prototype, the following issues were investigated:

1. Semantic annotations can improve the processing time and comprehension of the document.

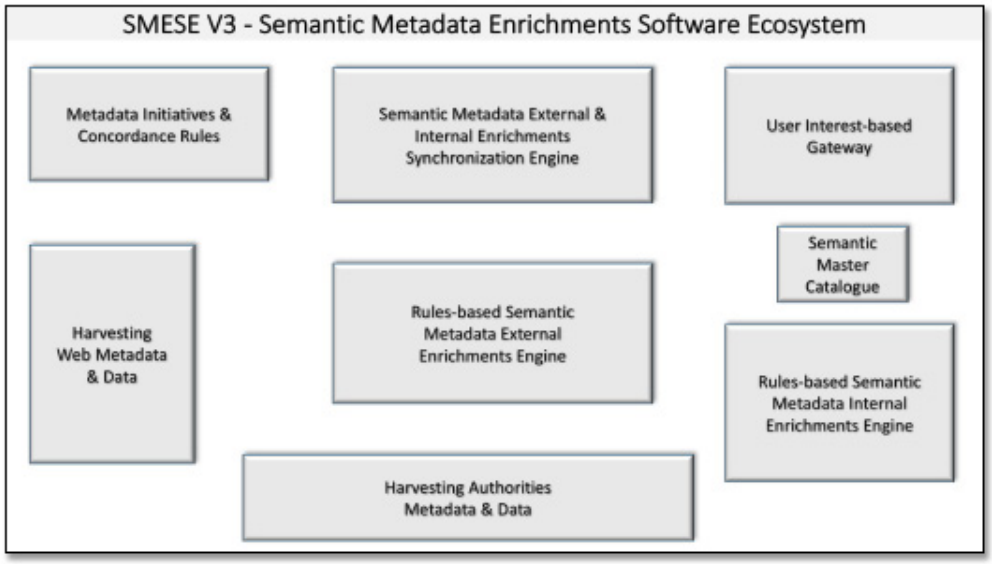


Figure 3. Semantic Enriched Metadata Software Ecosystem (SMESE) architecture.

2. Extending topic modeling into account co-occurrence to combine semantic relations and co-occurrence relations to complement each other.
3. Since latent co-occurrence relations between two terms cannot be measured in an isolated term-term view, the context of the term must be taken into account.
4. Use of machine learning techniques to allow the ecosystem SMESE to be able to find a new topic itself.

B. *Sentiment/Emotion Analysis*: The prototype developed has the following characteristics:

1. Traditional sentiment analysis methods mainly use terms and their frequency, parts of speech, rules of opinion and sentiment shifters; but semantic information is ignored in term selection.
2. Our contribution to sentiment analysis includes emotions.
3. The human contribution to improve the accuracy of our approach is taken into account.
4. Sentiment and emotion analysis are combined.
5. It is important to identify the sentiment and emotion of a book taking into account all the books of the collection.
6. The collection of documents and paragraphs are taken into account. In terms of granularity, most of the existing approaches are sentence-based.
7. These approaches did not take into account the surrounding context of the sentence which may cause some misunderstanding with discovery of sentiment/emotion. In our approach, the surrounding context of the sentence is included.

The prototype makes use of the proposed algorithm BM-SSEA (Semantic Sentiment and Emotion Analysis). The SMEE algorithm fulfills all the attributes of Table 2.

Table 2. SMESE characteristics.

1	Internal and external developers
2	Evaluative common technological platform
3	Controlled central part
4	Enable outside contributions and extensions
5	Variability-enabled architecture
6	Shared core assets
7	Automated and tool-supported product derivation
8	Outside contributions included in main platform
9	Social network and IoT integration
10	Semantic Metadata Internal Enrichments
11	Semantic Metadata External Enrichments
12	User Interest-based Gateway

The SMESE extends the SECO characteristics presented in [18] from 10 to 12. See Table 1 SECO characteristics versus Table 2 SMESE characteristics.

More specifically, the proposed SPLE approach is a combination of FORM and COPA approaches focusing on data and metadata enrichment. Through the combination of these two approaches, the following can be taken into account:

1. Administrative and organizational aspects such as roles and responsibilities, intergroup communication capabilities, personnel training, adoption of new technologies, strategic plans of the organization and marketing strategies.
2. Technical aspects such as requirements, design, implementation, test and maintenance.

With respect to CBSE, our SMESE includes a method for selecting composer components for design of an SPLE. This method can manage and control the complexities of the component selection problem in the creation of the declared product line. Also, the SMESE architecture supports runtime variability and multiple and dynamic binding times of products.

4. Subsystems within the SMESE Multi-Platform Architecture

The following sub-sections present in more detail the nine subsystems designed for the prototype of this SMESE architecture.

4.1. Metadata Initiatives & Concordance Rules

This section presents the details of the metadata initiatives & concordance rules, specifically the semantic metadata meta-catalogue (SMMC) as shown in Figure 2.

Metadata is structured information that describes, explains, locates, accesses, retrieves, uses, or manages an information resource of any kind. Metadata refers to data about data. Some use it to refer to machine understandable information, while others employ it only for records that describe electronic resources. In the library ecosystem, metadata is commonly used for any formal scheme of resource description, applying to any type of object, digital or non-digital. Many metadata schemes exist to describe various types of textual and non-textual objects including published books, electronic documents, archival documents, art objects, educational and training materials, scientific datasets and, obviously, the web.

Libraries and information centers are the intermediaries between the information, information sources and users. In order to make information accessible, libraries perform several activities, one of the most important and fundamental of which is cataloguing. The technological developments of the past 25 years have radically transformed both the process of cataloguing and access to information through catalogues.

Several rules have been proposed to cover the description and provision of access points for all library materials (entities). These rules are based on an individual framework for the description of library materials. There is no ecosystem

that allows the creation of universal, understandable and readable, metadata, that would describe all entities used in a library.

The most known metadata models are:

1. Dublin Core (DC): primarily designed to provide a simple resource description format for networked resources. DC does not have any coding to provide the necessary details for the specification of a record that could be converted to any machine readable coding like UNIMARC, MARC21.
2. UNIMARC: consists of data formulated by highly controlled cataloguing codes. This format is difficult to understand and unreadable for the end user. For this reason, MARC21 was proposed.
3. MARC21: is both flexible and extensible and allows users to work with data in ways specific to individual library needs. MARC21 remains difficult to understand, however.
4. RDF/RDA: mainly in Europe, is a new model that includes FRBRized Bibliographic Records.
5. BIBFRAME: mainly in North America, is a new model that includes FRBRized Bibliographic Records.

In addition, there is no mapping model among these that would make them interoperable. The overall challenge is to develop: (1) a modeling of partial international standardization of entities, (2) a modeling of partial international standardization of metadata, and (3) a modeling of partial international standardization of metadata mapping ontology.

Unfortunately, the power of metadata is limited: indeed, large national and international digital library projects, such as Europeana and the Digital Public Library of America, have highlighted the importance of sharing metadata across silos. While both of these projects have been successful in harvesting collections data, they have had problems with rationalizing the data and forming a coherent and semantic understanding of the aggregation.

In addition, organizations create digital collections and generate metadata in repository silos. Generally such metadata does not:

1. Connect the digitized items to their analogue sources.
2. Connect names to authority records (persons, organizations, places, etc.) nor subject descriptions to controlled vocabularies.
3. Connect to related online items accessible elsewhere.

Aggregators harvest this metadata that, in the process, generally becomes inaccurate. In fact, aggregators usually ignore idiosyncratic use of metadata schemas and enforce the use of designated metadata fields.

Connecting data across silos would help improve the ability of users to browse and navigate related entities without having to do multiple searches in multiple portals. The proposed model defines crosswalks that create pathways to different sources; each pathway checks the structure of the metadata source and then performs data harvesting. Figure 4 shows the SMMC model that addresses this issue.

In SMESE the metadata is classified into six categories:

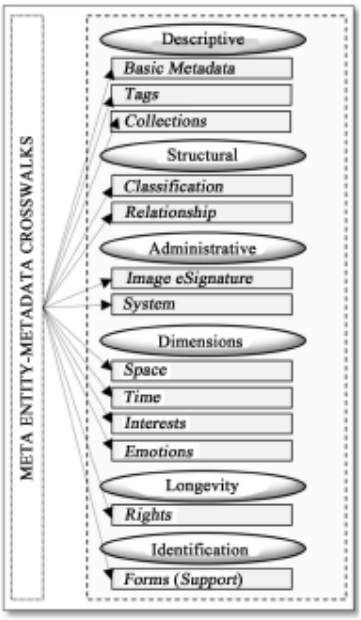


Figure 4. Semantic metadata meta-catalogue (SMMC).

1. *Descriptive metadata*: describes and identifies information resources at the local (system) level to enable searching and retrieving (e.g., searching an image collection to find paintings of animals) at the web-level, and to enable users to discover resources (e.g., searching the web to find digitized collections of poetry). Such metadata includes unique identifiers, physical attributes (media, dimensions, conditions) and bibliographic attributes (title, author/creator, language, keywords).
2. *Structural metadata*: facilitates navigation and presentation of electronic resources and provides information about the internal structure of resources (including page, section, chapter numbering, indexes, and table of contents) in order to describe relationships among materials (e.g., photograph B was included in manuscript A), and to bind the related files and scripts (e.g., File A is the JPEG format of the archival image File B).
3. *Administrative metadata*: facilitates both short-term and long-term management and processing of digital collections and includes technical data on creation and quality control, rights management, access control and usage requirements.
4. *Dimension, longevity and identification metadata*: are new classifications that aim to increase user satisfaction, in terms of expected interests and emotions. For example, dimension metadata regroups all metadata about space, time, emotions and interests. This metadata allows finding specific content. Another example: emotions may suggest specific content to a particular user at a

specific time and place. Furthermore, the source metadata identifies the provenance and the rights relative to the creation of the metadata.

4.2. Harvesting of Web Metadata & Data

The harvesting of web metadata & data sources such as:

1. Semantic digital resources
2. Digital resources
3. Portal/websites events
4. Social networks & events
5. Enrichment repositories
6. Discovery repositories

The integration of these sources in SMESE allows users to aggregate and enrich metadata and data.

4.3. Harvesting Authority Metadata & Data

This sub-section presents the details of the Harvesting of Authorities Metadata & Data.

The Semantic Multi-Platform Ecosystem consists of many authority sources, such as:

1. BAnQ (Bibliothèque et Archives nationales du Qc
2. BAC (Bibliothèque et Archives du Canada
3. BNF (Bibliothèque Nationale de France)
4. Library of Congress
5. British Library
6. Europeana
7. Spanish Library

The integration of these platforms in SMESE allows users to build an integrated authorities knowledge base.

4.4. Rules-Based Semantic Metadata External Enrichments Engine

This sub-section presents the details of the rule-based semantic metadata external enrichment engine.

Semantic searches over documents and other content types needs to use semantic metadata enrichment (SME) to find information based not just on the presence of words, but also on their meaning. It consists of:

1. Rule-based semantic metadata external enrichment engine.
2. Multilingual normalization.
3. Rule-based data conversion.
4. Harvesting metadata & data.

Linked open data (LOD) based semantic annotation methods are good candidates to enrich the content with disambiguated domain terms and entities (e.g. events, emotions, interests, locations, organizations, persons), see Figure 5, described through Unique Resource Identifiers (URIs) [46]. In addition, the original contents should be enriched with relevant knowledge from the respective

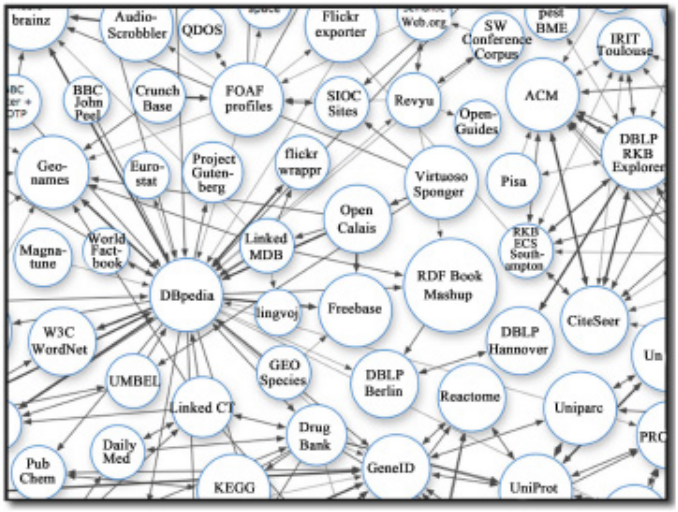


Figure 5. Linked Open Data (LOD).

LOD resources (e.g. that Justin Trudeau is a Canadian politician). This is needed to answer queries that require common-sense knowledge, which is often not present in the original content. For example: following semantic enrichment, a semantic search for events that provides specific emotions in Montreal according to individual interests this weekend would indeed provide relevant metadata about events in Montreal, even though not explicitly mentioned in the original content metadata.

The semantic annotation process of SMESE creates relationships between semantic models, such as ontologies and persons. It may be characterized as the semantic enrichment of unstructured and semi-structured contents with new knowledge and linking these to relevant domain ontologies/knowledge bases. It typically requires annotating a potentially ambiguous entity mention (e.g. Justin Trudeau) with the canonical identifier of the correct unique entity (e.g. depending on the content, http://dbpedia.org/page/Justin_Trudeau). The benefit of social semantic enrichment is that by surfacing annotated terms derived from the full-text content, concepts buried within the body of the paper/report can be highlighted. Also, the addition of terms affects the relevance ranking in full-text searches. Moreover, users can be more specific by limiting the search criteria to the subject or interest or emotion metadata (e.g. through faceted search).

4.5. Rule-Based Semantic Metadata Internal Enrichments Engine

This sub-section presents the details of the rule-based semantic metadata internal enrichment engine including software product line engineering (SPLE).

This sub-system includes:

1. A rule-based semantic metadata internal enrichment engine.

2. A multilingual normalization process.
3. Software Product Line Engineering (SPLE)
4. A topic, sentiment/emotion, abstract analysis and an automatic literature review.

These processes extract, analyze and catalogue metadata for topics and emotions involved in the SMESE ecosystem. These enrichment processes are based on information retrieval and knowledge extraction approaches. The text is analyzed making use of extension of text mining algorithms such as latent Dirichlet allocation (LDA), latent semantic analysis (LSA), support vector machine (SVM) and k-Means.

The different phases of the enrichment process by topics are:

1. Relevant and less similar documents selection phase.
2. Not annotated documents semantic term graph generation phase.
3. Topics detection phase.
4. Training phase.
5. Topics refining phase.

The different phases of the enrichment process by sentiments and emotions are:

1. Sentiment and emotion lexicon generation phase.
2. Sentiment and emotion discovery phase.
3. Sentiment and emotion refining phase.

One of the contributions of the SMESE for digital libraries is that it is not specific to one software product but can be applied to many products dynamically. In addition, it includes a semantic metadata enrichment (SME) process to improve the quality of search and discovery engines.

Indeed, our goal is to provide a SECO that offers a new way to share and learn knowledge. In practice, with the emergence of Big Data, knowledge is not easy to find at the right time and place. The proposed ecosystem uses an SPLE architecture that is a combination of FORM and COPA approaches to catalogue semantically different contents.

Furthermore, we introduce an SPLE decision support process (SPLE-DSP) in order to meet the SPLE characterization such as:

1. Runtime variability functionalities support.
2. Multiple and dynamic binding.
3. Context-awareness and self-adaptation.

SPLE-DSP supports the activation and deactivation of features and changes in the structural variability at runtime and takes into account automatic runtime reconfiguration according to different scenarios. In addition, SPLE-DSP rebinds to new services dynamically based on the description of the relationships and transitions between multiple binding times under an SPLE when the software adapts its system properties to a new context. To take into account context variability to model context-aware properties, SPLE-DSP makes use of an autonomous robot that exploits context information to adapt software behavior to varying conditions.

Furthermore, SPLE-DSP integrates the adaptation of assets and products dy-

namically. This helps products to evolve autonomously when the environment changes and provides self-adaptive and optimized reconfiguration. Additionally, SPLE-DSP exploits knowledge and context profiling as a learning capability for autonomic product evolution by enhancing self-adaptation.

The SPLE-DSP model is an optimized metadata based reconfiguration model where users select their preferences in terms of configuration of interests.

The dynamic and optimized metadata-based reconfiguration model (DOMRM) takes into account the preferences of several users who have distinct requirements in terms of desirable features and measurable criteria. For example:

1. In terms of hardware criteria, the user can select preferences in terms of memory and power consumption or feature attributes such as internet bandwidth or screen resolution.
2. In terms of software criteria, the user can select the entities and their properties, the property characteristics such as the displaying mode, and expected value type.

Indeed, when user preferences change at runtime, the system must be reconfigured to satisfy as many preferences as possible. Since user preferences may be contradictory, only some will be partially satisfied and a relevant algorithm needed to compute the most suitable reconfiguration. To overcome this drawback, we developed the use of a new metadata-based feature model, referred to as the BiblioMondo semantic feature model (BMSFM), to represent user preferences in terms of semantic features and attributes. Our BMSFM constitutes an evolution of traditional stateful feature models [51] that includes the set of user metadata based configurations in the model itself, which allows the representation of user decisions with attributes and cardinalities. More specifically, we developed a metadata-based reconfiguration model that defines all possible metadata and all possible entities that users may need in a specific domain. When a user needs new metadata, he uses the metadata-based request creation tool. The DOMRM model analyses the request and checks whether the requested metadata is relevant and does not already exist. Thus when needed the model automatically creates the new metadata and reconfigures the ecosystem which then becomes available for all users.

Figure 6 illustrates the DOMRM model we designed that is an optimized metadata based configuration for multiple users.

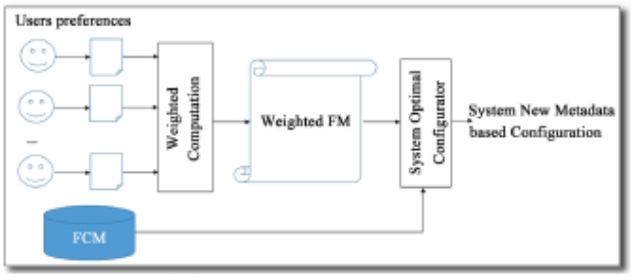


Figure 6. Optimized metadata based configuration for multiple users—DOMRM model.

When the user chooses preferences in terms of system behavior, the semantic weight of each feature is computed based on the feature configuration model (FCM). FCM represents the semantic relationship between features where each feature is active or not. In addition, FCM defines the rules that control the activation status of each feature according to its links with the other features. For example, a rule may be: feature F_i should never be activated when F_{i-1} is activated. Based on this rule, the model automatically activates or deactivates the feature.

The rules are also used to predict the behavior of the application based on the activation status of features according to user preferences. Notice that each user has his own weight per feature that is defined based on his use of the feature. This weight quantifies the importance of the feature for the user (more details about the DOMRM algorithm appear in Appendix A).

4.6. Semantic Metadata External & Internal Enrichments Synchronization Engine

This sub-section presents the semantic metadata external & internal enrichment synchronization engine which represents which processes to synchronize and which enrichments to push outside the ecosystem.

4.7. User Interest-Based Gateway

This sub-section presents the user interest-based gateway (UIG) that represents the person (mobile or stationary) who interacts with the ecosystem.

The users and contributors are categorized into five groups:

1. Interest-based gateway (mobile-first),
2. Semantic Search Engine (SSE),
3. Discovery,
4. Notifications,
5. Metadata source selection.

4.8. Semantic Master Catalogue

This sub-section presents the semantic master catalogue (SMC) that represents the knowledge base of the SMESE ecosystem.

5. An Implementation of SMESE for a Large Semantic Digital Library in Industry

The proposed SMESE architecture has been implemented for a large digital library. The product In Média V5 was implemented with a global metadata model defined with all the known entities and constraints. The catalogue contains more than 2 million items, with 18 entities and 132 defined metadata. SMMC identifies 1453 metadata and defines a metamodel that consists of a semantic classification of metadata into meta entities.

In addition to semantic web technologies, the characteristics and challenges of SMESE for large digital libraries are:

1. Automatic cataloguing with the least human intervention.
2. Metadata enrichment.
3. Discovery and definition of semantic relationships between metadata and records.
4. Semi-automatic classification of bibliographic records.
5. Semantic cataloging and validated metadata making use of a multilingual thesaurus.

First, we defined a list of entities, called Meta Entity, which introduced 193 items. These items represent all library materials. In addition, the structure of the model allows addition of new entities as may be required. Figure 7 shows the SMESE meta-entity model where for each entity there is: an ID, property Name, description, labels in different languages, and the domain that represents the logic group of the entity; for reason of formatting, Appendix C shows a readable version. The domain may be "user" as response value for a metadata. In this implementation, all instances of the entities of the domain can be the response value. The ID allows the user to uniquely identify the entity whatever the language, the source of entities or the metadata model (DC, UNIMARC, MARC21, RDA, BIBFRAME).

Next, the list of metadata is defined. 1341 metadata are defined. Each metadata entry has the following additional metadata called Meta Metadata: ID, related Content Type, is Enrichment, is Repeatable, thesaurus, type, and source Of Schema, which are defined as follows:

1. "source Of Schema" represents the origin.
2. "id" allows unique identification of the entity.
3. "property Name" is a comprehensive term that defines this metadata.
4. "UNIMARC", "MARC21", "property Name" allow users to create a mapping between them to make them interoperable.
5. "UNIMARC" and "MARC21" are codes such as 300$abcf.
6. "Expected type" represents the type of value that may be assigned to the metadata as response.
7. "isRelated" denotes that the response of the metadata is an entity where the identity is given by "related Content Type".
8. "thesaurus" mentions the thesaurus name that is used to control the metadata integrity.
9. "type" allows classification of the metadata as "descriptive", "structural", "administrative", "dimension", "longevity" or "identification".

This classification allows users to do meta research. Figure 8 shows an illustration of the Meta Metadata model; Appendix D shows a readable version.

The semantic matrix model is defined for each entity based on the metaentity and metadata model. This semantic matrix model allows users to define a metadata matrix for each entity where a metadata matrix denotes the logical subset of metadata of metadata model that describes a given entity. Figure 9 illustrates an example of a semantic metadata matrix for a specific content; Appendix E presents a readable version. The objective behind the matrix is to allow the reuse

META ENTITY														
Id.	Oldproperty Name	Property Name	Name	Description	BM label					Client Label		Source	Port Folio Entity Name	Domain
					Fr	En	Sp	Gr	Ne	Fr	En			
E1-n	Book	book	Book	C'est les contenus du type livre	Livre	Book	"	"	"	Livre	Book	BNF		bibliomondo
E2-n	Serial	serial	Serial	C'est les contenus du type périodique	Périodique	Serial	"	"	"	Périodique	Serial			bibliomondo
E3-n	Audio	audio	Audio	C'est les contenus du type audio	Audio	Audio	"	"	"	Audio	Audio			bibliomondo
E4-n		comic	Comic	C'est les contenus du type bande dessinée	Bande dessinée	Comic	"	"	"	Bande dessinée	Comic			bibliomondo
E5-n		Digital resource	Digital Resource	C'est les contenus du type ressource numérique	Ressource Numérique	Digital Resource	"	"	"	Ressource Numérique	Digital Resource			bibliomondo
E6-n		document	Document	C'est les contenus du type document	Document	Document	"	"	"	Document	Document			bibliomondo
E7-n		game	Game	C'est les contenus du type jeu	Jeu	Game	"	"	"	Jeu	Game			bibliomondo
E8-n	Image	image	Image	C'est les contenus de type Image	Image	Image	"	"	"	Image	Image	BNF		bibliomondo
E9-n	Musical Score	Musical score	Musical Score	C'est les contenus du type partition de musique	Partition de musique	Musical Score	"	"	"	Partition de musique	Musical Score			bibliomondo
E10-n	Video	video	Video	C'est les contenus du type vidéo	Vidéo	Video	"	"	"	Vidéo	Video			bibliomondo
E11-n	FRBR Work	Creative work	Creative work	The most generic kind of creative work, including books, movies, photographs, software programs, etc	Oeuvre	FRBRWork	"	"	"	Oeuvre	FRBRWor k	BNF		bibliomondo
E12-n		manifactation	Manifactation	C'est la manifestation d'une oeuvre	Manifestation	Manifestation	"	"	"	Manifesta-tion	Manifesta-tion			bibliomondo
E13-n		expression	Expression	C'est l' expression d'une manifestation	Expression	Expression	"	"	"	Expression	Expression			bibliomondo
E14-n		concept	Concept	C'est une concept	Concept	Concept	"	"	"	Concept	Concept			bibliomondo
E15-n		city	City	C'est les sous divisons d'un canton	Ville	City	"	"	"	Ville	City	BM		places
E16-n		Postal Address	Postal Address	C'est les contenus de type Adresse	Adresse postal	Postal address	"	"	"	Adresse postal	Postal address	BNF		places
E17-n		Place of interest	Place Of Interest	C'est un lieu qui suscite un intérêt particulié; par exemple, un site touristique, un musé, monument,...	Place d'intérêt	Place of interest	"	"	"	Place d'intérêt	Place of interest	BM		places
E18-n		country	Country	C'est un pays	Pays	Country	"	"	"	Pays	Country	BNF		places
E19-n		region	Region	C'est les sous divisions d'un pays; c'est le cas des régions en France, des États au États-Unis ou des provinve au Canada C'est les sous divisions d'un regions; par exemple les comtés au États unis ou les départements en France	Région	Region	"	"	"	Région	Region	BM		places

Figure 7. SMESE Meta Entity model.

META METADATA																											
Id.	Unimarc	Marc21	Old property Name	Property Name (DuoQ)	Property Name (Duo5)	Description	Bibl label					Client Label		Expected Type	Is Related	Related Content Type	Short Status	Enriched	Data Location	Hidden/Exportable	Thesaurus	Type id.U.S	Source Of Schema	Is Media Version
							Fr	En	Sp	Ne	Ge	Fr	En											
M1-m	200aebf	130/245	OPS245	OPS245	title	title	Titre	Title				Titre	Title	Text								Desc	BM	
M2-m	700abdg$l 710$abdgl 720af	0/9/100	OPS100	OPS100	author	author	Auteur	Author				Auteur	Author	Text			X		Local		Personnes ou Organisations	Desc	BNF	
M3-m	701abdg$l 711$abdgl 721af				creator	creator	Auteur	Author				Compositeur/ Interprète	Composer/ Performer	Text			X		Local		Personnes ou Organisations	Desc	BNF	
M4-m					promoter	promoter	Organisateur	Promoter				-	-	Text					Local		Personnes ou Organisations	Desc	BM	
M5-m	2$3$uz3	0/9/300	OPS300	OPS300	dcterms: description	description	A short description of the item.	Description	Description			Description	Description	Multiline Text					Local	X		Desc	BNF	
M6-m					Image	Image	An image of the item. This can be a image file or a URL or a fully described Image Object.	Imagette	Thumbnail			Imagette	Thumbnail	Image				X	BM.com	X		Desc	BNF	
M7-m	101$a	LOM146	LOM146		dcterms: language	language	C'est la langue (parlée ou écrite) dans laquelle le contenu est créé ou dans laquelle une personne (enseignant, par exemple, un livre rédigé en français sera cotée comme langue le français	Langue(s)	Language (s)			Langue(s)	Language(s)	Text					Local	X	Langues	Desc	BM	
M8-m	540$a	0/9/540	OPS540	OPS540	Title Added Uncontrolled Related	Title Added Uncontrolled Related	C'est d'autres titres du contenu qui pourraient être utilisés dans d'autres catalogues	Autres titres	Other titles			Autres titres	Other titles	Text					Local			Desc	BM	
M9-m	500abdlmnpstv	0/9/240	OPS240	OPS240	Title Uniform	Title Uniform	C'est le titre uniformé du contenu	Titre uniforme	Uniform title			Titre uniforme	Uniform title	Text					Local			Desc	BM	
M10-m						Also Available In	C'est les autres langues dans lesquelles le contenu a été traduit ou dans lesquelles une personne s'exprime	Disponible en d'autres langues	Available in other languages			Aussi disponible en	Also available in	Text					Local	X	Langues	Desc	HSC OIC	

Figure 8. SMESE metadata model.

of metadata for distinct entities. This extends the search range for entities, facilitates the search for users in terms of search criteria and increases the probability of achieving satisfying results.

After the definition of entities of collections and harvesting of metadata from the dispersed collections, a metadata crosswalk is carried out. This is a process in which relationships among the schema are specified, and a unified schema is developed for the selected collection. It is one of the important tasks for building "semantic interoperability" among collections and making the new digital library meaningful.

The most frequent issues regarding mapping and crosswalks are: incorrect mappings, misuse of metadata elements, confusion in descriptive metadata and administrative metadata, and lost information. Indeed, due to the varying degrees of depth and complexity, the crosswalks among metadata schemas may not-necessarily be equally interchangeable. To solve the issue of varying degrees

SEMANTIC MATRIX							Physical																						DM-n					
Id.	Uniname	Marc21	Old property Name	Property Name (SameQ)	Property Name (Based)	Description	Rdf Label Fr	Rdf Label En	Sp	De	Ge	Client Label Fr	Client Label En	Expected Type	Is Related	Related Content Type	Short Notice	Enriched	Data Location	Hidden	Repeatable	Thesaurus	Type	IsUrl	Source	Version IsMetdin
M004-m	0	0	0	0	Isnsin	It is the Biblio Mon de Social Identification Number, all the content has data metadata	ISMSIN	ISMSIN	0	0	0	ISMSIN	ISMSIN	Text	0	0	0	0	BM.com	0	0	0	0	0	BM	
M005-m	0	0	0	0	Date Of cataloging	It is the date of cataloging	Date de catalogage	Cataloging date	0	0	0	Date de catalogage	Cataloging date	Date	0	0	0	0	BM.com	0	0	0	0	0	BM	
M008-m	0	0	0	0	isni	International Standard Name Identifier	ISNI	ISNI	0	0	0	ISNI	ISNI	Text	0	0	0	0	BM.com	0	0	0	0	0	BM	
M0-m	78/3abcc2fgh/ 71/3abcc2dfgh/ 72/0ha0b			creator	creator	The creator/ author of this Creative Work. This is the same as the Author property for Creative Work	Auteur1	Author	0	0	0	Compositeur/ Interprète	Composer/ Performer	Text	0	0	X	0	Local	0	0	Personnes ou Organisations	Desc	0	BM	V3.1
M13-m	78/3abcc2fgh/ 71/2abcc2dfgh4/ 71/60ac57/22fa0b/ 7/30f0a4	OP67XX	O P67XX	contributor	contributor	C'est les autres personnes qui ont contribuées à la réalisation du contenu	Contributeurs	Contributors	0	0	0	Compositeur/ Auteurs/ Interprètes	Composers/ Songwriters/ Singers	Text	0	0	0	0	Local	0	0	Personnes ou Organisations	Desc	0	BM	0
M99-m	0/10ha	OP86XX	O P86XX	subject	subject	It is the subject of a content	Sujets	Subjects	0	0	0	Sujets	Subjects	Text	0	0	0	0	Local	0	X	Vedette Matiere	Admin	0	BM	V3.1
M321-m	0	0	0	Document Type	Document Type	C'est le type de document du contenu	Type de document	Document type	0	0	0	Type de document	Document type	List [type Documents]	0	0	0	0	Local	0	0	0	0	0	BM	V3.1
M322-m	0	0	0	Document Type Cluster	Document Type Cluster	C'est le groupe de type de document du contenu	Type de cluster de document	Cluster type of document	0	0	0	Type de cluster de document	Cluster type of document	Text	0	0	0	0	Local	0	0	0	0	0	BM	V3.1
M323-m	001	0	0	identifierInf	Identifier Dataset	C'est l'identificat bnf du contenu	Identifiant Bnf	Identifier Bnf	0	0	0	Identifier Bnf	Identifier Bnf	Text	0	0	0	0	Local	0	0	0	0	0	BM	V3.1
M130-m	0	0	0	Date Issued First	dateIssued First	C'est la date de la première édition du contenu	Date de la première édition	Date of first edition	0	0	0	Pre mière édition	First edition	Text	0	0	0	0	Local	0	0	0	Date	0	BM	V3.1
M158-m	0	0	0	Date Issued	Date Issued	C'est la date d'édition du contenu	Date Edition	Edition date	0	0	0	-	-	Date	0	0	0	0	Local	0	0	0	Date	0	BM	V3.1
M130-m	100	0	0	10kgl : date Of Publication Identification	Date Issued Formatted	Date of first broadcast/ publication	Date de publication	Date of publication	0	0	0	Date de publication	Date of publication	Date	0	0	0	0	Local	0	0	0	Date	0	BM	V3.1
M1-m	2000hasdefn1	OP9245	OP9245	title	title	It is the main title of a content	Titre	Title	0	0	0	Titre	Title	Text	0	0	0	0	0	0	0	0	Desc	0	BM	V3.1
M29-m	0	OP86XX	O P86XX	note	note	C'est les notes associées au contenu	Notes	Notes	0	0	0	Notes	Notes	Text	0	0	0	0	Local	0	0	0	Desc	0	BM	V3.1
M37-m	0/10ha	OP9828	OP9828	isbn	isbn	International Standard Book Number: il est possible de faire des entrées multiples	ISBN	ISBN	0	0	0	ISBN	ISBN	Text	0	0	0	0	Local	0	X	0	Desc	0	BM	V3.1

Figure 9. Example of a SMESE semantic matrix model.

of depth, we developed atomic metadata: these metadata allow description of the most elementary aspects of an entity. It then becomes easy to map all metadata from any schema.

Figure 10 illustrates a mapping ontology model where relationships are in red while simple descriptions are in black.

Figure 11 shows that each entity has at a minimum one source of schema denoted by the relationship "has Source" and a minimum of one metadata denoted by the relationship "has Metadata". The relationship "same As" is used to denote the mapping between distinct metadata or entity schema source.

The output of the ontology is an OWL file. This OWL file is used by a crosswalk to automatically assign metadata values that are harvested from distinct sources. In the proposed ecosystem two sources are harvested: Discogs (www.discogs.com) for music and Research Gate (www.researchgate.net) for academic papers.

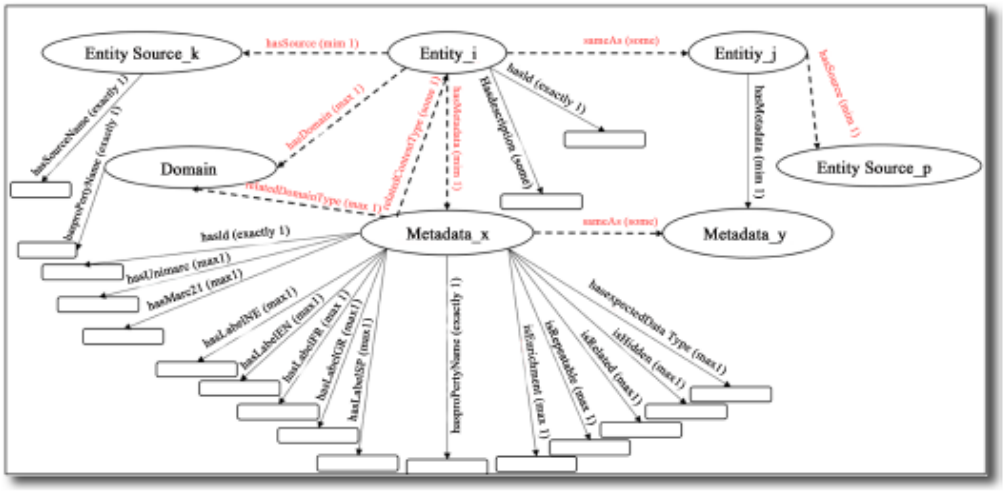


Figure 10. Ontology mapping model.

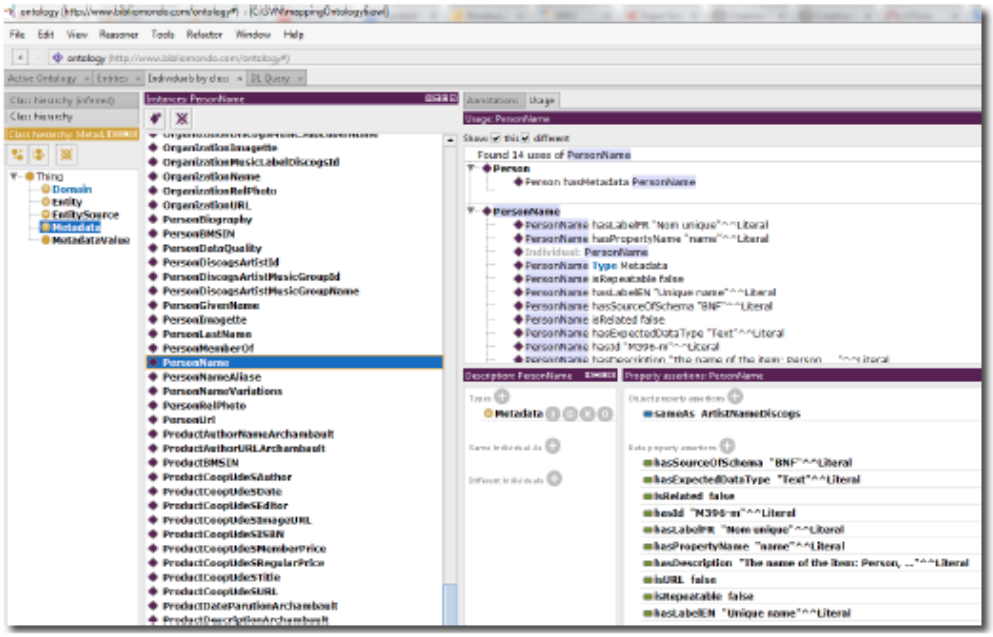


Figure 11. Ontology mapping implementation using Protégé.

A total of 94,015,090 metadata records were collected from these two sources:

1. From Discogs, we collected 7,983,288 entities: 2,621,435 music releases, 4,466,660 artists and 895,193 labels.
2. From researchGate, we collected 86,031,802 entities: 77,031,802 publications

and more than 9,000,000 researchers.

3. In fact, SMESE contains more than 3.4 billions triplets and growing.

6. Summary and Future Work

In this paper, we proposed a design and implementation of a semantic enriched metadata software ecosystem (SMESE).

The SMESE prototype, which was implemented at BiblioMondo, integrates data and metadata enrichment to support specific applications for distributed content management. To perform this integration, SMESE makes use of the software product line engineering (SPLE) approach, a component-based software development (CBSD) approach and our proposed new concept, called semantic metadata enrichment (SME) with distributed contents and mobile first design (MFD). In this implementation, the SPLE architecture is a combination of FORM and COPA approaches.

We also presented our implementation of SMESE for digital libraries. This included SPLE-DSP, a new decision support process for SPLE. SPLE-DSP consists of a dynamic and optimized metadata based reconfiguration model (DOMRM) where users select their preferences in the market place. SPLE-DSP takes into account runtime variability functionalities, multiple and dynamic binding, context-awareness and self-adaptation.

We also implemented the Meta Entity that represents all library materials and meta metadata. The ontology mapping model was then implemented to make our models interoperable with existing metadata models such as Dublin Core, UNIMARC, MARC21, RDF/RDA and BIBFRAME.

The major contributions of this paper are as follows:

1. Definition of a software ecosystem architecture (SMESE) that configures the application production process including software aspects based on CBSD and SPLE approaches.

a) The use of a LOD-based semantic enrichment model for semantic annotation processes.

b) The integration of National Research Council of Canada (NRC) emotion lexicon for emotion detection.

c) A repository of 43 thesaurus included in RAMEAU for semantical contextualization of concepts.

a. An extended latent Dirichlet allocation (LDA) algorithm for topic modeling.

2. Definition and partial implementation of semantic metadata enrichment using metadata SPLE and an SMMC (semantic master metadata catalogue) to create a universal metadata knowledge gateway (UMKG).

3. The design and implementation of an SMESE prototype of for a semantic digital library (Libēr).

This paper proposed a semantic metadata enrichments software ecosystem (SMESE) to support multi-platform metadata driven applications, such as a semantic digital library. Our SMESE integrates data and metadata based on mapping ontologies in order to enrich them and create a semantic master metadata

catalogue (SMMC).

Within the SPLE context, SPLE-DSP is used by SMESE to support dynamic reconfiguration. This consists of a dynamic and optimized metadata based reconfiguration model (DOMRM) where users select their preferences within the market place. SPLE-DSP takes into account runtime metadata-based variability functionalities, multiple and dynamic binding, context-awareness and self-adaptation. Our SMESE represents more than 200 million relationships (triplets).

Future work will include:

1. An enhanced ecosystem of connecting engines and rule-based algorithms to enrich metadata semantically, including topics and sentiments/emotions.
2. Evaluation of the performance of an implementation of the SMESE ecosystem using different projects, comparing results against existing techniques of metadata enrichments.

Exploring text summarization and automatic literature review as metadata enrichment, the semantic annotations could be used to enrich metadata and provide new types of visualizations by chaining documents backward and forward inside automated literature reviews.

References

[1] Lacasta, J., Nogueras-Iso, J., Falquet, G., Teller, J. and Zarazaga-Soria, F.J. (2013) Design and Evaluation of a Semantic Enrichment Process for Bibliographic Databases. *Data & Knowledge Engineering*, **88**, 94-107.

[2] Manikas, K. and Hansen, K.M. (2013) Software Ecosystems—A Systematic Literature Review. *Journal of Systems and Software*, **86**, 1294-1306.

[3] Christensen, H.B., Hansen, K.M., Kyng, M. and Manikas, K. (2014) Analysis and Design of Software Ecosystem Architectures—Towards the 4S Telemedicine Ecosystem. *Information and Software Technology*, **56**, 1476-1492.

[4] Shinozaki, T., Yamamoto, Y. and Tsuruta, S. (2015) Context-Based Counselor Agent for Software Development Ecosystem. *Computing*, **97**, 3-28. https://doi.org/10.1007/s00607-013-0352-y

[5] Jansen, S. and Bloemendal, E. (2013) Defining App Stores: The Role of Curated Marketplaces in Software Ecosystems. In: Herzwurm, G. and Margaria, T., Eds., *Software Business. From Physical Products to Software Services and Solutions: 4th International Conference, ICSOB* 2013, Potsdam, Germany, 11-14 June 2013, Springer, Berlin, Heidelberg, 195-206.

[6] Urli, S., Blay-Fornarino, M., Collet, P., Mosser, S. and Riveill, M. (2014) Managing a Software Ecosystem Using a Multiple Software Product Line: A Case Study on Digital Signage Systems. 40*th EUROMICRO Conference on Software Engineering and Advanced Applications*, Verona, 27-29 August 2014, 344-351. https://doi.org/10.1109/seaa.2014.23

[7] Albert, B.E., dos Santos, R.P. and Werner, C.M.L. (2013) Software Ecosystems Governance to Enable IT Architecture Based on Software Asset Management. 7*th IEEE International Conference on Digital Ecosystems and Technologies* (*DEST*), Menlo Park, CA, 24-26 July 2013, 55-60. https://doi.org/10.1109/dest.2013.6611329

[8] Mustl, J., Mustl, A. and Biffl, S. (2013) Elements of Software Ecosystem Early-Stage Design for Collective Intelligence Systems. *Proceedings of the* 2013 *International*

Workshop on Ecosystem Architectures, Saint Petersburg, 19 August 2013, 21-25. https://doi.org/10.1145/2501585.2501590

[9] da Silva Amorim, S., Almeida, E.S.D. and McGregor, J.D. (2013) Extensibility in Ecosystem Architectures: an Initial Study. *Proceedings of the* 2013 *International Workshop on Ecosystem Architectures*, Saint Petersburg, 19 August 2013, 11-15.

[10] Mens, T., Claes, M., Grosjean, P. and Serebrenik, A. (2014) Studying Evolving Software Ecosystems based on Ecological Models. In: Mens, T., Serebrenik, A. and Cleve, A., Eds., *Evolving Software Systems*, Springer, Berlin, Heidelberg, 297-326. https://doi.org/10.1007/978-3-642-45398-4_10

[11] dos Santos, R.P., Esteves, M.S., Freitas, G. and de Souza, J. (2014) Using Social Networks to Support Software Ecosystems Comprehension and Evolution. *Social Networking*, 3, 108-118. https://doi.org/10.4236/sn.2014.32014

[12] Robillard, M.P. and Walker, R.J. (2014) An Introduction to Recommendation Systems in Software Engineering. In: Robillard, P.M., Maalej, W., Walker, J.R. and Zimmermann, T., Eds., *Recommendation Systems in Software Engineering*, Springer, Berlin, Heidelberg, 1-11. https://doi.org/10.1007/978-3-642-45135-5_1

[13] Park, J.-G. and Lee, J. (2014) Knowledge Sharing in Information Systems Development Projects: Explicating the Role of Dependence and Trust. *International Journal of Project Management*, 32, 153-165.

[14] Lim, S.L., Bentley, P.J., Kanakam, N., Ishikawa, F. and Honiden, S. (2015) Investigating Country Differences in Mobile App User Behavior and Challenges for Software Engineering. *IEEE Transactions on Software Engineering*, 41, 40-64. https://doi.org/10.1109/TSE.2014.2360674

[15] Henderson-Sellers, B., Gonzalez-Perez, C., McBride, T. and Low, G. (2014) An Ontology for ISO Software Engineering Standards: 1) Creating the Infrastructure. *Computer Standards & Interfaces*, 36, 563-576.

[16] Di Ruscio, D., Paige, R.F., Pierantonio, A., Hutchinson, J., Whittle, J. and Rouncefield, M. (2014) Model-Driven Engineering Practices in Industry: Social, Organizational and Managerial Factors That Lead to Success or Failure. *Science of Computer Programming*, 89, 144-161.

[17] Ghapanchi, A.H., Wohlin, C. and Aurum, A. (2014) Resources Contributing to Gaining Competitive Advantage for Open Source Software Projects: An Application of Resource-Based Theory. *International Journal of Project Management*, 32, 139-152.

[18] Lettner, D., Angerer, F., Prahofer, H. and Grunbacher, P. (2014) A Case Study on Software Ecosystem Characteristics in Industrial Automation Software. *Proceedings of the* 2014 *International Conference on Software and System Process*, Nanjing, 26-28 May 2014, 40-49. https://doi.org/10.1145/2600821.2600826

[19] Gawer, A. and Cusumano, M.A. (2014) Industry Platforms and Ecosystem Innovation. *Journal of Product Innovation Management*, 31, 417-433. https://doi.org/10.1111/jpim.12105

[20] Andrés, C., Camacho, C. and Llana, L. (2013) A Formal Framework for Software Product Lines. *Information and Software Technology*, 55, 1925-1947.

[21] Metzger, A. and Pohl, K. (2014) Software Product Line Engineering and Variability Management: Achievements and Challenges. *Proceedings of the on Future of Software Engineering*, Hyderabad, 31 May-7 June 2014, 70-84.

[22] Harman, M., Jia, Y., Krinke, J., Langdon, W.B., Petke, J. and Zhang, Y. (2014) Search Based Software Engineering for Software Product Line Engineering: A Survey and Directions for Future Work. *Proceedings of the* 18*th International Software Product Line Conference*, Vol. 1, Florence, 15-19 September 2014, 5-18.

https://doi.org/10.1145/2648511.2648513

[23] Olyai, A. and Rezaei, R. (2015) Analysis and Comparison of Software Product Line Frameworks. *Journal of Software*, **10**, 991-1001. https://doi.org/10.17706/jsw.10.8.991-1001

[24] Capilla, R., Bosch, J., Trinidad, P., Ruiz-Cortés, A. and Hinchey, M. (2014) An overview of Dynamic Software Product Line Architectures and Techniques: Observations from Research and Industry. *Journal of Systems and Software*, **91**, 3-23.

[25] Krishnan, S., Strasburg, C., Lutz, R.R., Goseva-Popstojanova, K. and Dorman, K.S. (2013) Predicting Failure-Proneness in an Evolving Software Product Line. *Information and Software Technology*, **55**, 1479-1495.

[26] Quadri, A. and Abubakar, M. (2015) Software Quality Assurance in Component Based Software Development—A Survey Analysis. *International Journal of Computer and Communication System Engineering*, **2**, 305-315

[27] Singh, P.K., Sangwan, O.P., Singh, A.P. and Pratap, A. (2015) A Framework for Assessing the Software Reusability using Fuzzy Logic Approach for Aspect Oriented Software. *International Journal of Information Technology and Computer Science*, **7**, 12-20. https://doi.org/10.5815/ijitcs.2015.02.02

[28] Yadav, H.B. and Yadav, D.K. (2015) A Fuzzy Logic Based Approach for Phase-Wise Software Defects Prediction Using Software Metrics. *Information and Software Technology*, **63**, 44-57.

[29] Rettinger, A., Losch, U., Tresp, V., D'Amato, C. and Fanizzi, N. (2012) Mining the Semantic Web. *Data Mining and Knowledge Discovery*, **24**, 613-662. https://doi.org/10.1007/s10618-012-0253-2

[30] Jeremić, Z., Jovanović, J. and Gašević, D. (2013) Personal Learning Environments on the Social Semantic Web. *Semantic Web-Linked Data for Science and Education*, **4**, 23-51.

[31] Khriyenko, O. and Nagy, M. (2011) Semantic Web-Driven Agent-Based Ecosystem for Linked Data and Services. *3rd International Conferences on Advanced Service Computing*, Rome, 2011, 110-117.

[32] Lécué, F., Tallevi-Diotallevi, S., Hayes, J., Tucker, R., Bicer, V., Sbodio, M. and Tommasi, P. (2014) Smart Traffic Analytics in the Semantic Web with STAR-CITY: Scenarios, System and Lessons Learned in Dublin City. *Web Semantics: Science, Services and Agents on the World Wide Web*, **27-28**, 26-33.

[33] Ngan, L.D. and Kanagasabai, R. (2013) Semantic Web Service Discovery: State-of-the-Art and Research Challenges. *Personal and Ubiquitous Computing*, **17**, 1741-1752. https://doi.org/10.1007/s00779-012-0609-z

[34] Demir, K.A. (2015) Multi-View Software Architecture Design: Case Study of a Mission-Critical Defense System. *Computer and Information Science*, **8**, 12-31. https://doi.org/10.5539/cis.v8n4p12

[35] Aleti, A., Buhnova, B., Grunske, L., Koziolek, A. and Meedeniya, I. (2013) Software Architecture Optimization Methods: A Systematic Literature Review. *IEEE Transactions on Software Engineering*, **39**, 658-683. https://doi.org/10.1109/TSE.2012.64

[36] Ginters, E., Schumann, M., Vishnyakov, A. and Orlov, S. (2015) Software Architecture and Detailed Design Evaluation. *Procedia Computer Science*, **43**, 41-52.

[37] Yang, C., Liang, P. and Avgeriou, P. (2016) A Systematic Mapping Study on the Combination of Software Architecture and Agile Development. *Journal of Systems and Software*, **111**, 157-184.

[38] Oussalah, M., Bhat, F., Challis, K. and Schnier, T. (2013) A Software Architecture for Twitter Collection, Search and Geolocation Services. *Knowledge-Based Systems*,

37, 105-120.

[39] Capilla, R., Jansen, A., Tang, A., Avgeriou, P. and Babar, M.A. (2016) 10 Years of Software Architecture Knowledge Management: Practice and Future. *Journal of Systems and Software*, **116**, 191-205.

[40] de M. Neves, A.R., Carvalho, Á.M.G. and Ralha, C.G. (2014) Agent-Based Architecture for Context-Aware and Personalized Event Recommendation. *Expert Systems with Applications*, **41**, 563-573.

[41] Horcas, J.-M., Pinto, M. and Fuentes, L. (2016) An Automatic Process for Weaving Functional Quality Attributes Using A Software Product Line Approach. *Journal of Systems and Software*, **112**, 78-95.

[42] Ayala, I., Amor, M., Fuentes, L. and Troya, J.M. (2015) A Software Product Line Process to Develop Agents for the IoT. *Sensors*, **15**, 15640-15660. https://doi.org/10.3390/s150715640

[43] Mück, T.R. and Fröhlich, A.A. (2014) A Metaprogrammed C++ Framework for Hardware/Software Component Integration and Communication. *Journal of Systems Architecture*, **60**, 816-827.

[44] He, W. and Xu, L.D. (2014) Integration of Distributed Enterprise Applications: A Survey. *IEEE Transactions on Industrial Informatics*, **10**, 35-42. https://doi.org/10.1109/TII.2012.2189221

[45] Alférez, G.H., Pelechano, V., Mazo, R., Salinesi, C. and Diaz, D. (2014) Dynamic Adaptation of Service Compositions with Variability Models. *Journal of Systems and Software*, **91**, 24-47.

[46] Bontcheva, K., Kieniewicz, J., Andrews, S. and Wallis, M. (2015) Semantic Enrichment and Search: A Case Study on Environmental Science Literature. *D-Lib Magazine*, **21**, 1-18. https://doi.org/10.1045/january2015-bontcheva

[47] Krueger, R., Thom, D. and Ertl, T. (2015) Semantic Enrichment of Movement Behavior with Foursquare—A Visual Analytics Approach. *IEEE Transactions on Visualization and Computer Graphics*, **21**, 903-915. https://doi.org/10.1109/TVCG.2014.2371856

[48] Kunze, C. and Hecht, R. (2015) Semantic Enrichment of Building Data with Volunteered Geographic Information to Improve Mappings of Dwelling Units and Population. *Computers, Environment and Urban Systems*, **53**, 4-18.

[49] Fileto, R., Bogorny, V., May, C. and Klein, D. (2015) Semantic Enrichment and Analysis of Movement Data: Probably It Is Just Starting! *SIGSPATIAL Special*, **7**, 11-18. https://doi.org/10.1145/2782759.2782763

[50] Fileto, R., May, C., Renso, C., Pelekis, N., Klein, D. and Theodoridis, Y. (2015) The Baquara2 Knowledge-Based Framework for Semantic Enrichment and Analysis of Movement Data. *Data & Knowledge Engineering*, **98**, 104-122.

[51] Trinidad, P. (2012) Automating the Analysis of Stateful Feature Models. PhD Dissertation, University of Seville, Spain.

Appendix A: Dynamic and Optimized Metadata-Based Reconfiguration Model (DOMRM)

This Appendix presents the details of the DOMRM model. The main idea behind DOMRM is the more a user uses a specific feature, the more his weight for this feature increases. The weight *UjFi* of user *j* for feature *i* is given by:

$$UjFi = \frac{n(Uj, Fi)}{\sum_{k=1}^{P} n(Uk, Fi)} \tag{1}$$

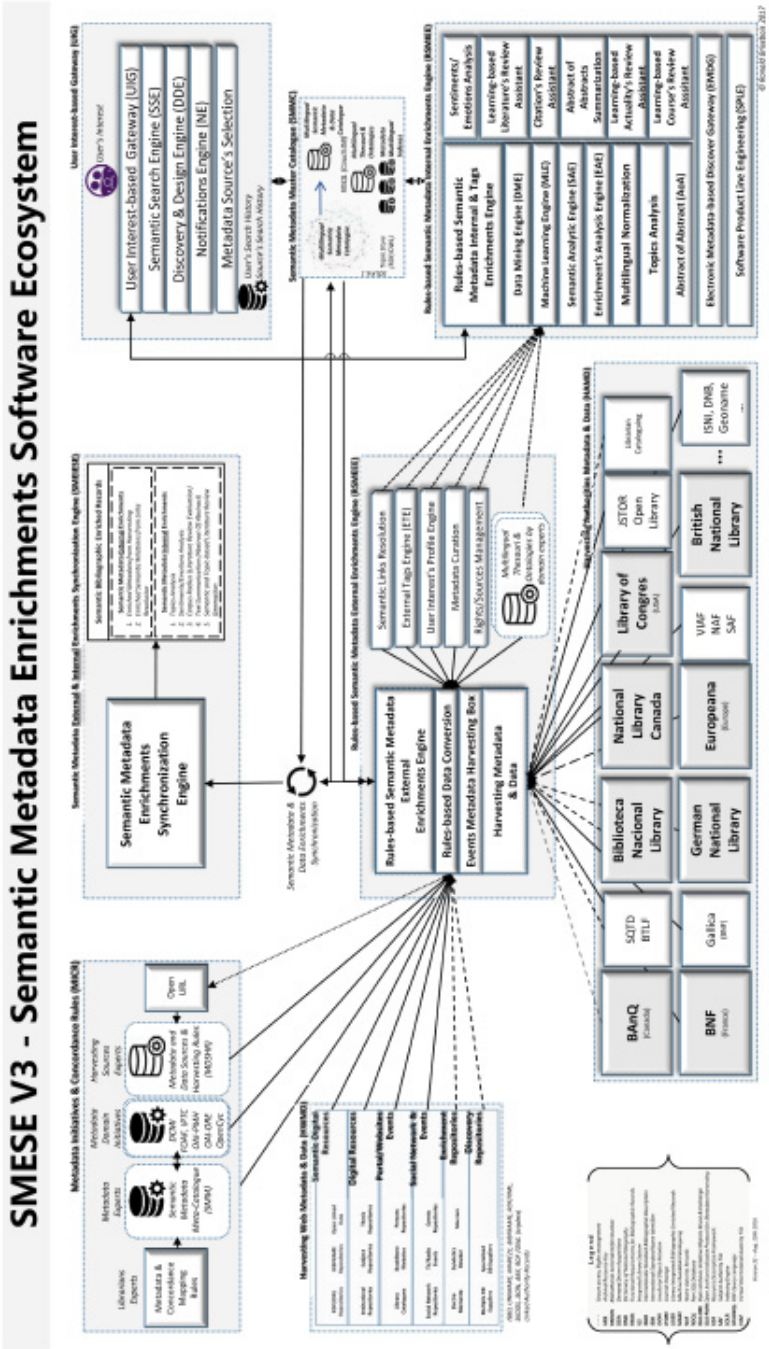
where $n(Uj, Fi)$ denotes the number of times user *j* used the feature *i*.

Making use of user weight per feature and their preferences, the feature weight that determines its activation or not is computed. Considering that *US* is the set of users who have selected a feature *Fi* (activation of feature), and *UR* is the set of users who have removed that feature (deactivation of feature), the value 1 is assigned when a user actives the feature, and −1 when he removes it. Let $c(Uj, Fi)$ be the choice of user j for the activation status of feature *Fi*. The weight of feature *Fi* can be defined using the following formula:

$$w(Fi) = \begin{cases} 1 & \text{whether } 0 < \sum_{Uk \in US\_UR} \left[c(Uk, Fi) \times UkFi \right] \\ -1 & \text{whether } 0 > \sum_{Uk \in US\_UR} \left[c(Uk, Fi) \times UkFi \right] \end{cases} \tag{2}$$

The computed weight of each feature allows one to define the weight FM that is used by the system optimal configurator with the FCM to generate the new configuration of the system for all users. When the feature weight is negative and the FIS rules allow de-activation, the feature is deactivated and when the feature weight is positive and the FIS rules allow activation the DOMRM model activates the feature. The activation status of the feature is not modified when the feature weight is null and the current activation status is conserved.

Appendix B: Proposed SMESE (Semantic Metadata Enrichments Software Ecosystem)



Appendix C: Figure 7. SMESE Meta Entity Model

META ENTITY														
Id.	Oldproperty Name	Property Name	Name	Description	BM label					Client Label		Source	Port Folio Entity Name	Domain
					Fr	En	Sp	Gr	Ne	Fr	En			
E1-n	Book	book	Book	C'est les contenus du type livre	Livre	Book	"	"	"	Livre	Book	BNF		bibliomondo
E2-n	Serial	serial	Serial	C'est les contenus du type périodique	Périodique	Serial	"	"	"	Périodique	Serial			bibliomondo
E3-n	Audio	audio	Audio	C'est les contenus du type audio	Audio	Audio	"	"	"	Audio	Audio			bibliomondo
E4-n		comic	Comic	C'est les contenus du type bande dessinée	Bande dessinée	Comic	"	"	"	Bande dessinée	Comic			bibliomondo
E5-n		Digital resource	Digital Resource	C'est les contenus du type ressource numérique	Ressource Numérique	Digital Resource	"	"	"	Ressource Numérique	Digital Resource			bibliomondo
E6-n		document	Document	C'est les contenus du type document	Document	Document	"	"	"	Document	Document			bibliomondo
E7-n		game	Game	C'est les contenus du type jeu	Jeu	Game	"	"	"	Jeu	Game			bibliomondo
E8-n	Image	image	Image	C'est les contenus de type Image	Image	Image	"	"	"	Image	Image	BNF		bibliomondo
E9-n	Musical Score	Musical score	Musical Score	C'est les contenus du type partition de musique	Partition de musique	Musical Score	"	"	"	Partition de musique	Musical Score			bibliomondo
E10-n	Video	video	Video	C'est les contenus du type vidéo	Vidéo	Video	"	"	"	Vidéo	Video			bibliomondo
E11-n	FRBR Work	Creative work	Creative work	The most generic kind of creative work, including books, movies, photographs, software programs, etc	Oeuvre	FRBRWork	"	"	"	Oeuvre	FRBRWor k	BNF		bibliomondo
E12-n		manifactation	Manifactation	C'est la manifestation d'une oeuvre	Manifestation	Manifestation	"	"	"	Manifesta-tion	Manifesta-tion			bibliomondo
E13-n		expression	Expression	C'est l' expression d'une manisfestation	Expression	Expression	"	"	"	Expression	Expression			bibliomondo
E14-n		concept	Concept	C'est une concept	Concept	Concept	"	"	"	Concept	Concept			bibliomondo
E15-n		city	City	C'est les sous divisons d'un canton	Ville	City	"	"	"	Ville	City	BM		places
E16-n		Postal Address	Postal Address	C'est les contenus de type Adresse	Adresse postal	Postal address	"	"	"	Adresse postal	Postal address	BNF		places
E17-n		Place of interest	Place Of Interest	C'est un lieu qui suscite un intérêt particulié; par exemple, un site touristique, un musé, monument,…	Place d'intérêt	Place of interest	"	"	"	Place d'intérêt	Place of interest	BM		places
E18-n		country	Country	C'est un pays	Pays	Country	"	"	"	Pays	Country	BNF		places
E19-n		region	Region	C'est les sous divisions d'un pays; c'est le cas des régions en France, des États au États-Unis ou des provinve au Canada	Région	Region	"	"	"	Région	Region	BM		places
				C'est les sous divisions d'un regions; par exemple les comtés au États unis ou les départements en France										

Appendix D: Figure 8. SMESE Metadata Model

META METADATA																											
								Bibl Label					Client Label														
Id.	Unimarc	Marc21	Old property Name	Property Name (DublQ)	Property Name (DublI)	Description	Fr	En	Sp	Ne	Gr	Fr	En	Expected Type	Is Related	Related Content Type	Short Notice	Enriched	Data Location	Hidden/Disponible	Thesaurus	Type inOrd.	Source Of Schema	In Media Version
M1-m	200aebd	130/245	OP0245	OP0245	title	title	It is the main title of a content.	Titre	Title			Titre	Title	Text								Desc	BM	
M2-m	700abgd 710abgd 720af	OP0100X	OP0100X	author	author	The author of this content. Please note that author is special in that HTML 1 provides a special mechanism for indicating authorship via the rel tag. That is equivalent to this and may be used interchangeably	Auteur	Author				Auteur	Author	Text			X		Local		Personnes ou Organisations	Desc	BNF	
M3-m	701abgd 711abgd 721af			creator	creator	The creator/author of this Creative Work. This is the same as the Author property for Creative Work	Auteur	Author				Compositeur/ Interprète	Composer/ Performer	Text			X		Local		Personnes ou Organisations	Desc	BNF	
M4-m					promoter	C'est le organisateur du contenu; c'est le cas des événements	Organisateur	Promoter				-	-	Text					Local		Personnes ou Organisations	Desc	BM	
M5-m	215ad	520/590	OP0590	OP0590	dcterms: description	description	A short description of the item.	Description	Description			Description	Description	Multiline Text					Local	X		Desc	BNF	
M6-m					Image	Image	An image of the item. This can be a image file or a URL or a fully described Image Object.	Imagette	Thumbnail			Imagette	Thumbnail	Image				X	BM.com	X		Desc	BNF	
M7-m	101$a	L0090$b	L0090$b	dcterms: language	language	C'est le langue (parlée ou écrite) dans laquelle le contenu est créé ou dans laquelle une personne comprenne par exemple, un livre rédigé en français sera contenu langue le français	Langue(s)	Language (s)				Langue(s)	Language(s)	Text					Local	X	Langues	Desc	BM	
M8-m	540ac	O0950$b	OP0540	Title Added Uncontrolled Related	Title Added Uncontrolled Related	C'est d'autres titres du contenu qui pourraient être utilisés dans d'autres catalogue	Autres titres	Other titles				Autres titres	Other titles	Text					Local			Desc	BM	
M9-m	500abgdmnpsrv	O0950$b	OP0240	Title Uniform	Title Uniform	C'est le titre modifié du contenu	Titre uniforme	Uniform title				Titre uniforme	Uniform title	Text					Local			Desc	BM	
M10-m					Also Available In	C'est les autres langues dans lesquelles le contenu a été traduit ou dans lesquelles une personne s'exprime	Disponible en d'autres langues	Available in other languages				Aussi disponible en	Also available in	Text					Local	X	Langues	Desc	BBC OIC	

Appendix E: Figure 9. Example of a SMESE Semantic Matrix Model

			SEMANTIC MATRIX						Physical											DN-n						
												META METADATA														
							Bbl label					Client Label														
Id.	Unimarc	Marc21	Old property Name	Property Name (BaseQ)	Property Name (Excel)	Description	Fr	En	Sp	Nr	Gr	Fr	En	Expected Type	Is Related	Related Content Type	Short Notice	Enriched	Data Location	Hidden	Repeatable	Thesaurus	Type	isUrl	Source	Version InMedia
M514-m	0	8	8	0	bmsin	It is the Biblio Mondo Serial Identification Numbers, all the content has this metadata	BMSIN	BMSIN	0	8	0	BMSIN	BMSIN	Text	8	0	0	8	BM.com	0	8	8	8	0	BM	
M850-m	0	8	8	0	Date Of cataloging	It is the date of cataloging	Date de catalogage	Cataloging date	0	8	0	Date de catalogage	Cataloging date	Date	8	0	0	8	BM.com	0	8	8	8	0	BM	
M860-m	0	8	8	0	isni	International Standard Name Identifier	ISNI	ISNI	0	8	0	ISNI	ISNI	Text	8	0	0	8	BM.com	0	8	8	8	0	BM	
M5-m	701#abcdfgh/ 711#abcdefgh/ 721#abfh			creator	creator	The creator/ author of this Creative Work. This is the same as the Author property for Creative Work	Auteur	Author	0	8	0	Compositeur/ Interprète	Composer/ Performer	Text	8	0	X	8	Local	0	8	Personnes ou Organisations	Desc	0	BNF	V3.1
M13-m	702#abcdfgh/ 712#abcdefgh4/ 716#acf/722#abf4/ 730#a4	OP07XX	OP07XX	contributor	contributor	C'est les autres personnes qui ont contribuées à la réalisation du contenu	Contributeurs	Contributors	0	8	0	Compositeurs/ Auteurs/ Interprètes	Composers/ Songwriters/ Singers	Text	8	0	0	8	Local	0	8	Personnes ou Organisations	Desc	0	BM	0
M96-m	610#a	OP60XX	OP60XX	subject	subject	It is the subject of a content	Sujets	Subjects	0	8	0	Sujets	Subjects	Text	8	0	0	8	Local	0	X	Vedette Matière	Admin	0	BM	V3.1
M321-m	0	8	8	Document Type	Document Type	C'est le type de document du contenu	Type de document	Document type	0	8	0	Type de document	Document type	List (type Document)	8	0	0	8	Local	0	8	8	8	0	BM	V3.1
M322-m	0	8	8	Document Type Cluster	Document Type Cluster	C'est le groupe de type de document du contenu	Type de cluster de document	Cluster type of document	0	8	0	Type de cluster de document	Cluster type of document	Text	8	0	0	8	Local	0	8	8	8	0	BM	V3.1
M323-m	001	8	8	identifierBnf	Identifier DataBnf	C'est l'identifiant bnf du contenu	Identifiant Bnf	Identifier Bnf	0	8	0	Identifier Bnf	Identifier Bnf	Text	8	0	0	8	Local	0	8	8	8	0	BM	V3.1
M150-m	0	8	8	Date Issued First	dateIssued First	C'est la date de la première édition du contenu	Date de la première édition	Date of first edition	0	8	0	Première édition	First edition	Text	8	0	0	8	Local	0	8	8	Dim	0	BM	V3.1
M158-m	0	8	8	Date Issued	Date Issued	C'est la date d'édition du contenu	Date Édition	Edition date	0	8	0	-	-	Date	8	0	0	8	Local	0	8	8	Dim	0	BM	V3.1
M153-m	100	8	8	edtf : date Of Publication Manifestation	Date Issued Formatted	Date of first broadcast/ publication	Date de publication	Date of publication	0	8	0	Date de publication	Date of publication	Date	8	0	0	8	Local	0	8	8	Dim	0	BM	V3.1
M1-m	200#acdehi	OP0245	OP0245	title	title	It is the main title of a content	Titre	Title	0	8	0	Titre	Title	Text	8	0	0	8	0	0	8	8	Desc	0	BM	V3.1
M29-m	0	OP05XX	OP05XX	note	note	C'est les notes associées au contenu	Notes	Notes	0	8	0	Notes	Notes	Text	8	0	0	8	Local	0	8	8	Desc	0	BM	V3.1
M37-m	010#a	OP0020	OP0020	isbn	isbn	International Standard Book Number. Il est possible de faire des entrées multiples	ISBN	ISBN	0	8	0	ISBN	ISBN	Text	8	0	0	8	Local	0	X	8	Desc	0	BM	V3.1

Paper 2:

A Semantic Metadata Enrichment Software Ecosystem based on Metadata and Affinity Models

Ronald Brisebois, Alain Abran, Apollinaire Nadembega

I.J. Information Technology and Computer Science, 2017, 05, 1-16
Published Online May 2017 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2017.04.01

A Semantic Metadata Enrichment Software Ecosystem based on Metadata and Affinity Models

Ronald Brisebois
École de technologie supérieure, University of Quebec, Montreal, Canada
Email: ronald.brisebois.1@ens.etsmtl.ca

Alain Abran[1] and Apollinaire Nadembega[2]
[1]École de technologie supérieure, University of Quebec, Montreal, Canada
Email: alain.abran@etsmtl.ca
[2] Network Research Lab., University of Montreal, Montreal, Canada
Email: Apollinaire.nadembega@umontreal.ca

***Abstract*—**Information systems need to be more flexible and to allow users to find content related to their context and interests. Metadata harvesting and metadata enrichments could represent a way to help users to find content and events according to their interests. However, metadata are underused and represents an interoperability challenge. This paper presents a new framework, called SMESE, and the implementation of its prototypes that consists of its semantic metadata model, a mapping ontology model and a user interest affinity model. This proposed framework makes these models interoperable with existing metadata models.

SMESE also proposes a decision support process supporting the activation and deactivation of software features related to metadata. To consider context variability into account in modeling context-aware properties, SMESE makes use of an autonomous process that exploits context information to adapt software behavior using an enhanced metadata framework. When the user chooses preferences in terms of system behavior, the semantic weight of each feature is computed. This weight quantifies the importance of the feature for the user according to their interests.

This paper also proposed a semantic metadata analysis ecosystem to support data harvesting according to a metadata model and a mapping ontology model. Data harvesting is coupled with internal and external enrichments. The initial SMESE prototype represents more than 400 millions of relationships (triplets). To conclude, this paper also presents the design and implementation of different prototypes of SMESE applied to digital ecosystems.

***Index Terms*—**Metadata, metadata enrichment, metadata model, ontology, semantic metadata enrichment, software ecosystem.

I. Introduction

With more and more data available on the web, how users search and discover content or events is of crucial importance. There is growing research on interaction paradigms investigating how users may benefit from (1) the expressive power of semantic web standards, (2) the existing cataloguing models and metadata enrichments.

The semantic web may be defined as the transformation of the world wide web to a database of semantic linked resources, where data may be widely reused and shared [1]. Semantic information discovery approaches [2, 3] are now challenging traditional keyword-based information retrieval methods. This retrieval problem is further burdened by the poor quality of the metadata content in many digital collections.

Software ecosystems (SECO) [4-19] are defined as the interaction of a set of actors on top of a common technological ecosystem providing a number of software interactions or webservices [4, 5]. In SECO, internal and external actors create and compose relevant solutions together with a community of domain experts and users to satisfy customer requirements. This poses new challenges since the software systems are being evolved by various distributed development teams, communities, experts and technologies.

There is growing agreement on the main characteristics of SECO, including a common technological platform enabling outside contributions and variability-enabled architectures. Nine characteristics have been identified [20] that focus on technical processes for system development, interconnection and evolution.

Gawer and Cusumano [21] have analyzed a wide range of industry examples of SECO and identified two predominant types of platforms:

1. Internal platforms: defined as a set of assets organized in a common structure from which a company can efficiently develop and produce a stream of derivative products.
2. External platforms: defined as products, services, or technologies that act as a foundation upon which external innovators, organized as an innovative business ecosystem, can develop their own complementary products, technologies, or services.

Concurrently modern software demands more and more adaptive features. The semantic web [22-26] and linked data are some of the most important concepts to support Semantic Metadata Enrichment (SME) in a SECO architecture [27-33].

Today, semantic web technologies offer a new level of flexibility, interoperability and a way to enhance peer communication and knowledge sharing. Indeed, a semantic web engine, ensures more closely relevant results based on the ability to understand the definition and user-specific meaning of the word or term being searched for. Semantic search engines try to understand the context in which the words are being used, resulting in more relevant results with greater user satisfaction.

However, to enrich web data by transforming them into knowledge that may be more accessible and understandable by systems and users, this paper proposes a framework using metadata model architecture, referred as the SMESE framework (Semantic Metadata Enrichment Software Ecosystem).

The SMESE architecture includes semantic metadata enrichment engines based on a metadata model, a mapping ontologie model and a user interest affinity model. It integrates and enriches metadata.

SMESE also proposes a decision support process supporting the activation and deactivation of software features related to metadata. To consider context variability into account in modeling context-aware properties, SMESE makes use of an autonomous process that exploits context information to adapt software behavior using an enhanced metadata framework.

The multi-platform metadata model of SMESE was presented in [34] while this paper focuses specifically on the metadata and affinity models of SMESE.

The remainder of the paper is organized as follows. Section 2 presents the related works. Section 3 summarizes the multi-platform framework of the proposed SMESE, and Section 4 presents the related eight metadata and affinity models and sub-systems of SMESE. Section 5 presents the prototype of SMESE implementation in an industry context. Section 6 presents a summary and ideas for future work.

II. Related Works

This related works section is at the intersection of SECO and SME and presents the three related research axes:

1. SECO architecture using component integration.
2. SECO architecture and concepts.
3. Semantic metadata enrichments (SME).

The related works section is at the intersection of SECO and SME. First, the SECO architecture is presented, second, the concept and finally the semantic metadata enrichments.

A. SECO architecture using components integration

Software ecosystems [4-6, 12, 21, 28, 32] consist of multiple software products, often interrelated to each other by means of dependency relationships. When one product undergoes changes and issues a new release, this may or may not lead other products to upgrade their dependencies. Unfortunately, the upgrade of a component may create a series of issues. In their systematic literature review of SECO research, Manikas and Hansen [4] report that:

1. There is little consensus on what is a SECO.
2. Few analytical models of SECO exist.
3. Little research is done in the context of real-world.

They define a SECO as the interaction of a set of actors on top of a common technological platform. They also identify three main perspective in a SECO architecture:

1. Software engineering: the focus is on technical issues related directly or indirectly to the technology platform.
2. Business and management: the focus is on the business, organizational and management aspects.
3. Relationships: represent the social aspect.

B. SECO architecture and concept

Christensen, Hansen, Kyng and Manikas [5] define the concept of SECO architecture as a set of structures comprised of actors and software elements, the relationships among them, and their properties.

Demir [27] also proposes a software architecture that is strongly related to a defence system and limited to military personnel. Their multi-view of the SECO architecture is described step by step.

Neves, Carvalho and Ralha [33] propose an architectural solution based on ontology and the spreading algorithm that offers personalized and contextualized event recommendations in the university domain. They use an ontology to define the domain knowledge model and the spreading activation algorithm to learn user patterns through discovery of user interests.

Alferez, Pelechano, Mazo, Salinesi and Diaz [35] propose a framework that uses semantically rich variability models at runtime to support the dynamic adaptation of service compositions. They propose that service compositions be abstracted as a set of features in a variability model.

C. Semantic metadata enrichment

Bontcheva, Kieniewicz, Andrews, and Wallis [36] investigate semantic metadata automatic enrichment and search methods. In particular, the benefits of enriching articles with knowledge from linked open data resources are investigated with a focus on the environmental science domain. They also propose a form-based semantic search interface to facilitate environmental science researchers in carrying out better semantic searches. Their proposed model is limited to linking terms with DBpedia URI and does not take into account the semantic meaning of terms.

Some authors focus their enrichment model on person mobility trace data [37-40]. Krueger, Thom, and Ertl [37] show how semantic insights can be gained by enriching trajectory data with place of interest (POI) information using social media services. They handle semantic uncertainties in time and space, which result from noisy, imprecise, and missing data, by introducing a POI decision model in combination with highly interactive visualizations.

Kunze and Hecht [38] propose an approach to processing semantic information from user-generated OpenStreetMap (OSM) data that specifies non-residential use in residential buildings based on OSM attributes, so-called tags, which are used to define the extent of non-residential use.

The conclusions from these related works are:

1. Metadata-based architecture needs to be flexible and meet administrative, organizational and technical aspects.
2. Several proposed models do not take into account autonomic mechanisms to guide the self-adaptation of service compositions according to changes in the computing infrastructure.
3. There is no SECO architecture that takes into account several semantic enrichment aspects.
4. Current metadata and entity enrichment models are limited to only one domain for their semantic enrichment process and therefore do not involve several enriched metadata and entity models.
5. Current metadata and entity enrichment models do not take into account person mobility trace data gathering and analysis in the enrichment process of metadata.

III. SMESE ARCHITECTURE

This section presents the architecture of the proposed Semantic Metadata Enriched Software Ecosystem (SMESE). It is based on metadata semantic internal and external enrichments and their interoperability. Each component of the SMESE architecture is based on semantic metadata to generate, extract, discover and enrich metadata based on mapping ontologies and a user interest affinity model. SMESE makes use of contents and linked data analysis.

For the new generation of information and data management, metadata is one of the most efficient material for data aggregation and understanding. For example, it is easier to find a specific set of interests for users based on metadata such as content topics, or based on the sentiments expressed in a content. Furthermore, it is possible to increase user satisfaction by reducing the user interest gap using appropriate metadata. To make this feasible, content and events need to be semantically enriched. In other words, to achieve specific searches, specific metadata must be available including semantic topics, sentiments and abstracts. However, at the present time and according to our prototype, more than 85% of the content does not have these metadata.

The SMESE prototypes include an engine to aggregate multiple catalogues or datasets from the web, libraries, universities, bookstores, #tag collections, museums, and cities. Central indexes typically include: full text and citations from publishers, full text and metadata from open source collections, full text, abstracting and indexing from aggregators and subscription databases. They are in different formats and are also called either base index, unified index, or foundation index.

The SMESE framework enhances bibliographic records with semantic metadata enrichments. It searches and discovers actual collections or novelties, including: works, books, DVDs, CDs, comics, games, pictures, videos peoples, legacy collections, organizations, rewards, TVs, radios, museums and other events calendar. The prototype creates triplets to define relationships enriching metadata's content. To be able to map the user interest and the content metadata, the prototype includes a user interest affinity model. This model (see Fig. 1) includes:

1. An algorithm to recommend to user contents or events matching his interest according to the user interest affinity model;
2. An algorithm to rank dynamically the contents or events according to the user interest affinity model;

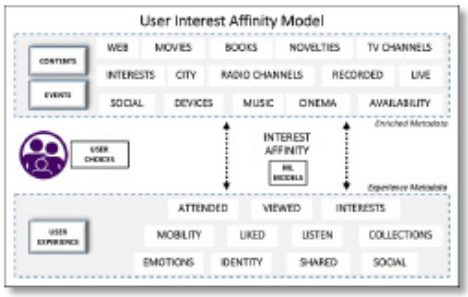


Fig. 1. User Interest Affinity Model.

Semantic relationships between the contents, persons, organization and places are defined and curated in the master metadata catalogue. Topics, sentiments and emotions are extracted automatically from the contents but with respect to their context. The average library has hundreds of thousands of catalogue records waiting to be transformed into linked data, turning those thousands of records into millions of relationships (triplets).

SMESE must allow users to find topically related content through an interest-based search and discovery engine. Transforming bibliographic records into semantic data is a complex problem that includes interpreting and transforming the information. Many international organizations have partly done this heavy work and already have much bibliographic metadata converted into triple-stores but there is not a definition of a common catalogue using the same semantic metadata model for all standards.

The SMESE prototypes harvest and analyse multiple catalogues and linked open data (LOD) from libraries, universities, bookstores, #tag collections, museums, open catalogues, national catalogues to produce semantic metadata enrichments.

Central indexes typically include full text and citations from publishers, full texts, abstracting and indexing from aggregators and subscription databases, and different formats (such as MARC) from library catalogues.

The SMESE framework allows to connect bibliographic records and semantic metadata enrichments (SEM) into a unified master metadata catalogue. The next figure (Fig. 2) presents the four levels of the metadata enrichment view uses par SMESE: (1) Meta-Entity (black), (2) Entity (blue), (3) Semantic metadata enrichment (grey), and (4) Contents & Events (white).

Fig. 2. Metadata enrichment view.

Semantic relationships between content, persons, organizations, events and places are defined and curated in the master metadata catalogue. Topics and sentiments are extracted (where possible) from the content, its context and related objects.

Recent catalogues support the ability to publish and search collections of descriptive entities (described by a list of generic metadata) for data, content and related information objects. Metadata in catalogues represent resource characteristics that can be indexed, queried and displayed by both humans and machine. Enriched catalogue metadata are needed to support the discovery and notification of information within an information community.

SMESE includes an automated approach for semantic metadata enrichment that allows users to perform interest-based semantic search or discovery more efficiently. To summarize, SMESE makes the following contributions:

A. ***Architecture, prototype and analysis of SMESE*** - Semantic Metadata Enrichment Software Ecosystem. (See Fig. 3 Detailed of the ecosystem; Appendix A shows a more readable version).

This new semantic ecosystem SMESE has the ability to harvest and enrich bibliographic records externally (from the web) and internally (from text data). The main components of the ecosystem are (see Fig. 3 and Appendix A shows a readable version):

1. Metadata initiatives & concordance rules
2. Harvesting web metadata & data
3. Harvesting authority metadata & data
4. Rule-based semantic metadata external enrichments
5. Rule-based semantic metadata internal enrichments
6. Semantic metadata external & internal enrichment synchronization
7. User interest-based gateway
8. Semantic master catalogue

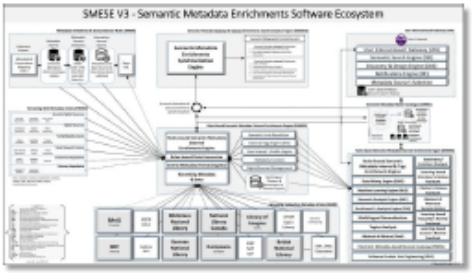


Fig. 3. Detailed Semantic Enriched Metadata Software Ecosystem [34].

B. ***Topic detection/generation*** - A prototype was developed to automate the generation of topics from the text of a document using our algorithm SATD (Semantic Annotation-based Topic Detection). In this research prototype, the following issues were investigated:

1. Semantic annotations can improve the processing time and comprehension of the document.
2. Extending topic model into account co-occurrence to combine semantic relations and co-occurrence relations to complement each other.
3. Since latent co-occurrence relations between two terms cannot be measured in an isolated term-term view, the context of the term must be taken into account.
4. Use of machine learning techniques to allow the SMESE ecosystem to be able to find a new topic itself.

C. ***Sentiment Analysis*** - The prototype developed has the following characteristics:

1. Traditional sentiment analysis methods mainly use terms and their frequency, parts of speech, rules of opinion and sentiment shifters; SMESE use semantic information to perform his analysis.
2. The collection of documents and paragraphs are taken into account. In terms of granularity, most of the existing approaches are sentence-based.
3. In SMESE prototypes, the surrounding context of the sentence is included. The traditional approaches do not take into account the surrounding context of the sentence which may cause some misunderstanding with discovery of sentiment.

The prototype makes use of the proposed algorithm SSEA (Semantic Sentiment and Emotion Analysis). This algorithm fulfills all the attributes of Table 1.

The SMESE extends the SECO characteristics presented in [20] from number 10 to 12 [34]. See Table 1 SECO characteristics versus SMESE characteristics.

Table 1. SMESE characteristics [34].

Number	Model	Characteristics	
1	SECO	Internal and external developers	
2	SECO	Evaluative common technological platform	
3	SECO	Controlled central part	
4	SECO	Enable outside contributions and extensions	
5	SECO	Variability-enabled architecture	
6	SECO	Shared core assets	
7	SECO	Automated and tool-supported product derivation	
8	SECO	Outside contributions included in the main platform	
9	SECO	Social network and IoT integration	
10	SMESE	Semantic Metadata Internal Enrichments	X
11	SMESE	Semantic Metadata External Enrichments	X
12	SMESE	User Interest Affinity Model	X

IV. Subsystems within the SMESE Architecture

The following sub-sections present in more detail the eight subsystems designed for the prototype of the SMESE architecture.

A. Metadata initiatives & concordance rules

This sub-section presents the details of the Metadata initiatives & concordance rules, specifically the semantic metadata meta-catalogue as shown in Fig. 3.

Metadata is a structured information that describes, explains, locates, accesses, retrieves, uses, or manages an information resource of any kind. Metadata refers to data about data. Some use metadata to refer to machine understandable information, while others employ it only for records that describe electronic resources. In the library ecosystem, metadata is commonly used for any formal scheme of resource description, applicable to any type of object, digital or non-digital. Many metadata schemes exist to describe various types of textual and non-textual objects including published books, electronic documents, archival documents, art objects, educational and training materials, scientific datasets and, obviously, the web.

Actually there is no common meta-model that allows the creation of universal, understandable and readable meta-model, that would describe all entities used in all the libraries.

The most popular metadata models are:

1. Dublin Core (DC): primarily designed to provide a simple resource description format for networked resources.
2. UNIMARC: consists of data formulated by highly controlled cataloguing codes.
3. MARC21: is both flexible and extensible and allows users to work with data in ways specific to individual library needs.
4. RDF/RDA: mainly in Europe, it includes FRBR capability.
5. BIBFRAME: mainly in North America, it includes FRBR capability.

There is no known mapping model among these that would make them interoperable. The overall challenge is to prototype: (1) a meta model of partial international standardization of entities, (2) a model of partial metadata mapping ontology and user interest affinity model.

In addition, organizations create digital collections and generate metadata in repository silos. In general, such metadata does not:

1. Connect the digitized items to their analogue sources.
2. Connect names to authority records (persons, organizations, places, etc.) nor subject descriptions to controlled vocabularies.
3. Connect to related online items accessible elsewhere.

Aggregators harvest this metadata that, in the process, generally becomes inaccurate. Indeed, aggregators usually ignore idiosyncratic use of metadata schemas and enforce the use of designated metadata fields.

Connecting data across silos would help improve the ability of users to browse and navigate related entities without having to do multiple searches in multiple portals from different catalogues. The proposed model defines crosswalks that create pathways to different sources; each pathway checks the structure of the metadata source and then performs data harvesting. Fig. 4 shows the semantic metadata model that SMESE propose to address these issues.

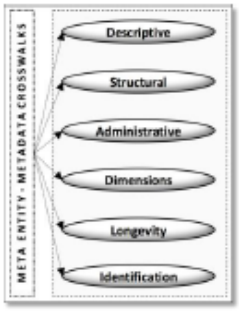


Fig. 4. Semantic metadata model.

In SMESE the proposed metadata are classified into six different categories:

1. *Descriptive metadata:* describes and identifies information resources at the local (system) level to enable searching and retrieving at the web-level, and to enable users to discover resources. Such metadata includes unique identifiers, physical attributes (media, dimensions, conditions) and bibliographic attributes (title, author/creator, language, keywords).
2. *Structural metadata:* facilitates navigation and presentation of electronic resources and provides information about the internal structure of resources (including page, section, chapter numbering, indexes, and table of contents) in order to describe relationships among metadata and entities.
3. *Administrative metadata:* facilitates both short-term and long-term management and processing of digital collections and includes technical data on creation and quality control, rights management, access control and usage requirements.
4. *Dimension metadata:* is a new classification that aim to increase user satisfaction, in terms of expected interests and emotions. Dimension metadata regroups all metadata about space, time, emotions and interests. Another example: emotions may suggest specific content to a particular user at a specific time and place. Furthermore, the source identifies the provenance and the rights relative to the creation of the metadata.
5. *Longevity metadata:* is a new classification that aim to manage the rights related to the content (entity).
6. *Identification metadata:* is a new classification that aim to manage the type of form or support of the media that contains the content (entity).

Semantic searches over documents and other content types needs to use semantic metadata enrichment (SME) to find information based not just on the presence of words, but also on their meaning. LOD based semantic annotation methods are good candidates to enrich the content with disambiguated domain terms and entities (e.g. events, emotions, interests, locations, organizations, persons), described through Unique Resource Identifiers (URIs) [36]. In addition, International Standard Names Identifier (ISNI) is proposed by National Libraries to organize and catalogue the semantic metadata relationships, see Fig. 5 adapted from the source [41] where the symbol with three blue dots represents a semantic repository using triplets. The BNF is identifying workflows with publishers to provide them with ISNIs for new authors. ISNI represents the opportunity to help to enrich an author's metadata and the quality of the authority files. ISNI Semantic relationships allow to connect together many sources of information such as:

1. Wikipedia,
2. Wikidata,
3. Union List of Artist Names,
4. IdRef,
5. Data.bnf.fr,
6. BNF Catalog,
7. SNAC,
8. AGORHA,
9. VIAF,
10. Data.banq.ca.

Fig. 5 shows also the introduction of ISNI semantic relationships into the semantic metadata meta-catalogue of the SMESE prototype.

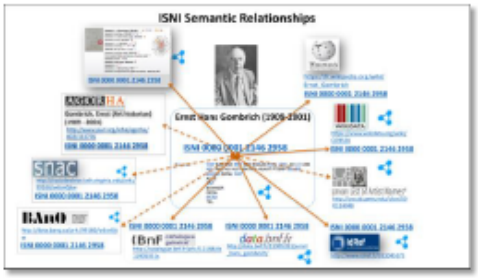


Fig. 5. ISNI semantic relationships of semantic metadata meta-catalogue in the SMESE prototype (adapted from [41]).

The original contents should be enriched with relevant knowledge from the respective LOD resources. This is needed to answer queries that require common-sense knowledge, which is often not present in the original content. For example: following semantic enrichment, a semantic search for events that provide specific emotions (e.g., happiness, joy, etc.) in Montreal according to individual interests this weekend would provide relevant metadata about events in Montreal, even though not explicitly mentioned in the original content metadata.

The semantic annotation process of SMESE creates relationships between semantic models, such as ontologies and persons. It may be characterized as the semantic enrichment of unstructured and semi-structured contents with new knowledge and linking these to relevant domain ontologies/knowledge bases. This requires the usage of ISNI, or other authority files or other techniques.

These processes extract, analyze and catalogue metadata for topic and sentiments involved in the SMESE ecosystem. As of today, 5 million records (entity) have been harvested over a potential target of close to 500 million, see Table 2 for an overview of the detail about harvested metadata and data (p.e. papers and events) in the prototype. For each content type many metadata and data have been extracted and enriched. These enrichment processes are based on information retrieval and knowledge extraction approaches. The text is analyzed by means of extensions of text mining algorithms such as latent Dirichlet allocation (LDA), latent semantic analysis (LSA), support vector machine (SVM) and k-Means.

Table 2. Harvesting statistics related to metadata

No	URL Sources	Status	%	Total Content	Total harvested
1	https://www.academia.edu/	h	1	15 363 735	142 217
2	https://www.researchgate.net/	h	0	100 000 000	27 984
3	http://www.opendoar.org/	h	0	235 828 824	385 155
4	http://www.amazon.com/	h	1	4 703 063	44 322
5	http://www.paulines.qc.ca/	h	76	171 120	130 000
6	http://exedre.leslibrairies.ca/	h	63	171 120	107 680
7	http://www.fnac.com/	h	67	156 224	104 109
8	http://www.librairiepoirier.ca/	f	100	176 162	176 162
9	http://www.archambault.ca/	f	100	165 405	165 405
10	http://www.renaudbray.com/accueil.aspx	f	100	347 380	347 380
11	http://www.usherbrooke.coop/	f	100	47 412	47 412
12	http://www.biblairie.qc.ca/	f	100	213	213
13	http://librairiemediaspaul.ca	f	100	29 938	29 938
14	https://www.leslibraires.fr/	f	100	888 750	888 750
15	http://www.mollat.com/	f	100	505 729	505 729
TOTAL				465 379 760	2 967 439

Status : f: finished and h: harvesting

SMESE is not specific to one software product but can be applied to many products dynamically. In addition, it includes a semantic metadata enrichment (SME) process to improve the quality of search and discovery engines.

The proposed SMESE framework uses an SPLE architecture that is a combination of FORM and COPA to catalogue semantically different contents.

SMESE also proposes a decision support process called SPLE-DSP. It supports the activation and deactivation of software features related to metadata and takes into account automatic runtime reconfiguration according to different scenarios. To take context variability into account in modeling context-aware properties, SPLE-DSP makes use of an autonomous process that exploits context information to adapt software behavior using a generic metadata model.

When the user chooses preferences in terms of system behavior, the semantic weight of each feature is computed based on the software feature configuration model (FCM). FCM represents the semantic relationship between features where each feature is active or not. In addition, FCM defines the rules that control the activation status of each feature according to its links with other features. For example, a rule may be: feature Fi should never be activated when Fi-1 is activated. Based on this rule, the FCM automatically activates or deactivates the feature.

The rules are also used to predict the behavior of the application based on the activation status of features according to users' selections. Note that individual users have their own weight per feature, defined on the basis of that user's use of the feature. This weight quantifies the importance of the feature for the user.

B. Harvesting of web metadata & data

The harvesting of web metadata & data sources such as:

1. Semantic digital resources
2. Digital resources
3. Portal/websites events
4. Social networks & events
5. Enrichment repositories
6. Discovery repositories

The integration of these sources in SMESE allows users to aggregate and enrich metadata.

C. Harvesting authority metadata & data

This sub-section presents the details of the Harvesting of Authority Metadata & Data are presented in Fig. 6.

Fig. 6. Harvesting of authority metadata & data.

The integration of these authority sources in SMESE allows users to build an integrated authorities knowledge base.

D. Rules-based semantic metadata external enrichments engine

This sub-section presents the details of the rule-based semantic metadata external enrichment engine included in SMESE.

Semantic searches over documents and other content types needs to use semantic metadata enrichment (SME) to find information based not just on the presence of

words, but also on their meaning and context. The rule-based semantic metadata external enrichment engine consists of:

1. Rule-based semantic metadata external enrichments.
2. Multilingual normalization.
3. Rule-based data conversion.
4. Harvesting metadata & data.

Semantic annotation methods are good candidates to enrich the content with disambiguated domain terms and entities (e.g. events, sentiments, interests, locations, organizations and persons), described through Unique Resource Identifiers (URIs) [36]. In addition, the original contents should be enriched with relevant knowledge from the respective linked open data resources (e.g. that Barack Obama is an American politician or Justin Trudeau is a Canadian politician). This is needed to answer queries that require common-sense knowledge, which is often not present in the original content. For example: following semantic enrichment, a semantic search for events that provides specific emotions (e.g., happiness, joy) in New York (or another city) according to individual interests this weekend would indeed provide relevant metadata about events in New York (or another city), even though not explicitly mentioned in the original content metadata. Furthermore, the linguistic aspect (context) of the knowledge is critical to analyse the metadata and corresponding data or content.

The semantic annotation process of SMESE creates relationships between semantic models, such as ontologies and persons. It may be characterized as the semantic enrichment of unstructured and semi-structured contents with new knowledge and linking these to relevant domain ontologies/knowledge bases. It typically requires annotating a potentially ambiguous entity mention with the canonical identifier of the correct unique entity. The benefit of social semantic enrichment is that by surfacing annotated terms derived from the full-text content, concepts buried within the body of the paper/report can be highlighted. Also, the addition of terms affects the relevance ranking in full-text searches. Moreover, users can be more specific by limiting the search criteria to the subject or interest or emotion metadata (e.g. through a faceted search).

E. Rule-based semantic metadata internal enrichments engine

This sub-section presents the details of the rule-based semantic metadata internal enrichment engine. This sub-system includes:

1. A rule-based semantic metadata internal enrichment engine.
2. A topic, sentiment/emotion, abstract analysis and an automatic literature review.

These processes extract, analyze and catalogue metadata for topics and sentiments involved in the SMESE ecosystem. These enrichment processes are based on information retrieval and knowledge extraction approaches. The text is analyzed making use of extension of text mining algorithms such as latent Dirichlet allocation, latent semantic analysis, support vector machine and k-Means. The different phases of the enrichment process by sentiments and emotions are:

1. Sentiment and emotion lexicon generation phase.
2. Sentiment and emotion discovery phase.
3. Sentiment and emotion refining phase.

One of the contributions of the SMESE is that it is not specific to one software product but can be applied to many products dynamically. In addition, it includes two semantic metadata enrichment (SME) processes to improve the quality of search and discovery engines; the external process who analyses the context of the data while harvesting and the internal process who analyses the content of the data.

F. Semantic metadata external & internal enrichments synchronization engine

This sub-section presents the semantic metadata external & internal enrichment synchronization engine which represents which processes to synchronize and which enrichments to push outside the ecosystem. Mainly this engine has the objective to find out the new content and context from the last harvesting.

G. User interest-based gateway

This sub-section presents the user interest-based gateway that represents the person (mobile or stationary) who interacts with the SMESE ecosystem. This engine use the metadata created by SMESE to give better results or recommendation to the users. The users and contributors are categorized into five groups:

1. Interest-based gateway
2. Semantic Search Engine
3. Discovery
4. Notifications
5. Metadata source selection.

H. Semantic master catalogue

This semantic master catalogue (SMC) represents the knowledge base of the SMESE ecosystem based on his evolving meta model of metadata. The SMC aggregates all triplets and their relationships created by the engines of SMESE. SMC includes also all the thesaurii and ontologies for a specific domain of interest.

V. AN IMPLEMENTATION OF SMESE FOR DIGITAL ECOSYSTEMS

The proposed SMESE architecture has been implemented for some digital ecosystems. The SMESE prototypes implement partially its metadata model and framework. The catalogue contains more than 2 million items, with 18 entities and 132 defined metadata. One of the prototype identifies 1453 metadata and defines a semantic classification.

First, we defined a list of entities, called Meta Entity, which introduced 193 items. These items represent all library materials. The structure of the model allows addition of new entities as may be required. The domain may be 'user' as response value for a metadata. In this implementation, all instances of the entities of the domain

can be the response value. 1341 metadata have been defined.

This classification allows users to search content according to their interests. Fig. 7 shows an illustration of the Metadata model; Appendix B shows a readable version.

Fig. 7. SMESE prototype metadata model.

The semantic matrix model is defined for each entity based on the meta entity and metadata model. This semantic matrix model allows users to define a metadata matrix for each entity where a metadata matrix denotes the logical subset of metadata of metadata model that describes a given entity. Fig. 8 illustrates an example of a semantic metadata matrix for a specific content; Appendix C presents a readable version. The objective behind the matrix is to allow the reuse of metadata for distinct entities.

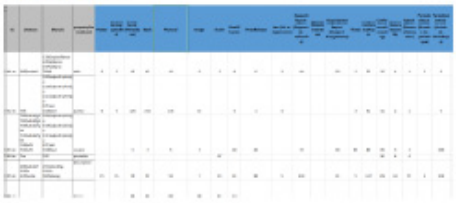


Fig. 8. Entities of a SMESE semantic matrix model.

After the definition of entities of collections and harvesting of metadata from the dispersed collections, a metadata crosswalk is carried out. This is a process in which relationships among the schema are specified, and a unified schema is developed for the selected collection.

The most frequent issues regarding mapping and crosswalks are: incorrect mappings, misuse of metadata elements, confusion in descriptive metadata and administrative metadata, and lost information. Indeed, due to the varying degrees of depth and complexity, the crosswalks among metadata schemas may not - necessarily be equally interchangeable. To solve the issue of varying degrees of depth, we developed atomic metadata: these metadata allow description of the most elementary aspects of an entity. It then becomes easy to map all metadata from any schema.

This OWL file from the ontology is used by a crosswalk to automatically assign metadata value that are harvested from distinct sources.

A total of 94,015,090 metadata records were collected from these different sources:

1. From Discogs (www.discogs.com) for music, we collected 7,983,288 entities: 2,621,435 music releases, 4,466,660 artists and 895,193 labels.
2. From ResearchGate (www.researchgate.net) for academic papers, we collected 86,031,802 entities: 77,031,802 publications and more than 9,000,000 researchers.
3. From academia (www.academia.edu) for academic papers, we collected 143,277 entities: 135,101 publications and more than 8,175 researchers.
4. From TV hebdo (www.tvhebdo.com) for TV channel programs, we collected 268,147,499 entities: 385 TV channel and 268,147,114 TV programs.
5. From OpenDOAR (www.opendoar.org) for scientific contents, we collected 235,828,824 entities: 96,265,327 thesis and 139,563,497 publications.

SMESE now contains more than 4.3 billion triplets and is growing.

VI. SUMMARY AND FUTURE WORK

In this paper, we proposed a design and implementation of SMESE, a semantic enriched metadata software ecosystem including a user interest affinity model. The SMESE prototype, integrates data and metadata enrichment to support internal and external metadata enrichments.

SMESE also includes a decision support process. It supports the activation and deactivation of software features related to metadata. To take context variability into account in modeling context-aware properties, SMESE makes use of an autonomous process that exploits context information to adapt software behavior using a generic metadata model. When the user chooses preferences in terms of system behavior, the semantic weight of each feature is computed based on the software feature configuration model. Individual users have their own weight per feature, defined on the basis of that user's use of the feature. This weight quantifies the importance of the feature for the user according to their interests.

We also presented our implementation of SMESE including the semantic metadata model. The ontology mapping model was then implemented to make the models interoperable with existing metadata models.

This paper proposed a semantic metadata enrichments software ecosystem to support multi-platform metadata driven applications. SMESE integrates data and metadata based on mapping ontologies in order to enrich them and create a semantic master metadata catalogue. SMESE prototype represents more than 400 million relationships (triplets).

The major contributions of this paper are as follows:

1. Definition of a metadata-based software ecosystem:
   a. Enhancing the SECO characteristics from 9 to 12.

b. The use of a LOD-based semantic enrichment model for semantic annotation processes.
c. A repository of 43 thesaurii included in RAMEAU for semantical contextualization of concepts.
d. An extended latent Dirichlet allocation algorithm for topic analysis.

2. Prototype of SMESE ecosystem for harvesting data and metadata and generating semantic metadata enrichments.
3. Prototype of a user interest affinity model.
4. The design and implementation of an SMESE prototype for different standards in digital ecosystems.

Future work related to SMESE ecosystem will include:

1. Some enhancements to be able to enrich metadata semantically, including evolving user interest.
2. Further evaluations of the affinity model with different prototype and datasets.

Exploring text summarization and automatic literature review as metadata enrichments. The semantic annotations could be used to enrich metadata and provide further data to improve the user interest affinity model.

REFERENCES

[1] J. Lacasta, J. Nogueras-Iso, G. Falquet, J. Teller, and F. J. Zarazaga-Soria, "Design and evaluation of a semantic enrichment process for bibliographic databases," *Data & Knowledge Engineering*, vol. 88, pp. 94-107, 2013. doi:http://dx.doi.org/10.1016/j.datak.2013.10.001

[2] Junaid Rashid, and M. W. Nisar, "A Study on Semantic Searching, Semantic Search Engines and Technologies Used for Semantic Search Engines," *International Journal of Information Technology and Computer Science(IJITCS)*, vol. 8, no. 10, pp. 82-89, 2016. doi:http://dx.doi.org/10.5815/ijitcs.2016.10.10

[3] Hepsiba Mabel V, and J. Selwyn, "A Review on the Knowledge Representation Models and its Implications," *International Journal of Information Technology and Computer Science(IJITCS)*, vol. 8, no. 10, pp. 72-81, 2016. doi:http://dx.doi.org/10.5815/ijitcs.2016.10.09

[4] K. Manikas, and K. M. Hansen, "Software ecosystems – A systematic literature review," *Journal of Systems and Software*, vol. 86, no. 5, pp. 1294-1306, 2013. doi:http://dx.doi.org/10.1016/j.jss.2012.12.026

[5] H. B. Christensen, K. M. Hansen, M. Kyng, and K. Manikas, "Analysis and design of software ecosystem architectures – Towards the 4S telemedicine ecosystem," *Information and Software Technology*, vol. 56, no. 11, pp. 1476-1492, 2014. doi:http://dx.doi.org/10.1016/j.infsof.2014.05.002

[6] T. Shinozaki, Y. Yamamoto, and S. Tsuruta, "Context-based counselor agent for software development ecosystem," *Computing*, vol. 97, no. 1, pp. 3-28, 2015. doi:http://dx.doi.org/10.1007/s00607-013-0352-y

[7] S. Jansen, and E. Bloemendal, "Defining App Stores: The Role of Curated Marketplaces in Software Ecosystems," *Software Business. From Physical Products to Software Services and Solutions: 4th International Conference, ICSOB 2013, Potsdam, Germany, June 11-14, 2013. Proceedings*, G. Herzwurm and T. Margaria, eds., pp. 195-206, Berlin, Heidelberg: Springer Berlin Heidelberg, 2013. doi:http://dx.doi.org/10.1007/978-3-642-39336-5_19

[8] S. Urli, M. Blay-Fornarino, P. Collet, S. Mosser, and M. Riveill, "Managing a Software Ecosystem Using a Multiple Software Product Line: A Case Study on Digital Signage Systems." pp. 344-351, 2014. doi:http://dx.doi.org/10.1109/SEAA.2014.23

[9] B. E. Albert, R. P. d. Santos, and C. M. L. Werner, "Software ecosystems governance to enable IT architecture based on software asset management." pp. 55-60, 2013. doi:http://dx.doi.org/10.1109/DEST.2013.6611329

[10] J. Musil, A. Musil, and S. Biffl, "Elements of software ecosystem early-stage design for collective intelligence systems," in Proceedings of the 2013 International Workshop on Ecosystem Architectures, Saint Petersburg, Russia, 2013, pp. 21-25. doi:http://dx.doi.org/10.1145/2501585.2501590

[11] S. d. S. Amorim, E. S. D. Almeida, and J. D. McGregor, "Extensibility in ecosystem architectures: an initial study," in Proceedings of the 2013 International Workshop on Ecosystem Architectures, Saint Petersburg, Russia, 2013, pp. 11-15. doi:http://dx.doi.org/10.1145/2501585.2501588

[12] T. Mens, M. Claes, P. Grosjean, and A. Serebrenik, "Studying Evolving Software Ecosystems based on Ecological Models," *Evolving Software Systems*, T. Mens, A. Serebrenik and A. Cleve, eds., pp. 297-326, Berlin, Heidelberg: Springer Berlin Heidelberg, 2014. doi:http://dx.doi.org/10.1007/978-3-642-45398-4_10

[13] R. dos Santos, P. , M. Esteves, S. , G. Freitas, and J. de Souza, "Using Social Networks to Support Software Ecosystems Comprehension and Evolution," *Social Networking*, vol. 3, no. 2, pp. 108-118, 2014. doi:http://dx.doi.org/10.4236/sn.2014.32014

[14] M. P. Robillard, and R. J. Walker, "An Introduction to Recommendation Systems in Software Engineering," *Recommendation Systems in Software Engineering*, P. M. Robillard, W. Maalej, J. R. Walker and T. Zimmermann, eds., pp. 1-11, Berlin, Heidelberg: Springer Berlin Heidelberg, 2014. doi:http://dx.doi.org/10.1007/978-3-642-45135-5_1

[15] J.-G. Park, and J. Lee, "Knowledge sharing in information systems development projects: Explicating the role of dependence and trust," *International Journal of Project Management*, vol. 32, no. 1, pp. 153-165, 2014. doi:http://dx.doi.org/10.1016/j.ijproman.2013.02.004

[16] S. L. Lim, P. J. Bentley, N. Kanakam, F. Ishikawa, and S. Honiden, "Investigating Country Differences in Mobile App User Behavior and Challenges for Software Engineering," *IEEE Transactions on Software Engineering*, vol. 41, no. 1, pp. 40-64, 2015. doi:http://dx.doi.org/10.1109/TSE.2014.2360674

[17] B. Henderson-Sellers, C. Gonzalez-Perez, T. McBride, and G. Low, "An ontology for ISO software engineering standards: 1) Creating the infrastructure," *Computer Standards & Interfaces*, vol. 36, no. 3, pp. 563-576, 2014. doi:http://dx.doi.org/10.1016/j.csi.2013.11.001

[18] D. Di Ruscio, R. F. Paige, A. Pierantonio, J. Hutchinson, J. Whittle, and M. Rouncefield, "Model-driven engineering practices in industry: Social, organizational and managerial factors that lead to success or failure," *Science of Computer Programming*, vol. 89, pp. 144-161, 2014. doi:http://dx.doi.org/10.1016/j.scico.2013.03.017

[19] A. H. Ghapanchi, C. Wohlin, and A. Aurum, "Resources contributing to gaining competitive advantage for open source software projects: An application of resource-based theory," *International Journal of Project Management*, vol. 32, no. 1, pp. 139-152, 2014. doi:http://dx.doi.org/10.1016/j.ijproman.2013.03.002

[20] D. Lettner, F. Angerer, H. Prahofer, and P. Grunbacher, "A case study on software ecosystem characteristics in industrial automation software," in Proceedings of the 2014 International Conference on Software and System Process, Nanjing, China, 2014, pp. 40-49. doi:http://dx.doi.org/10.1145/2600821.2600826

[21] A. Gawer, and M. A. Cusumano, "Industry Platforms and Ecosystem Innovation," *Journal of Product Innovation Management*, vol. 31, no. 3, pp. 417-433, 2014. doi:http://dx.doi.org/10.1111/jpim.12105

[22] A. Rettinger, U. Losch, V. Tresp, C. D'Amato, and N. Fanizzi, "Mining the Semantic Web," *Data Min. Knowl. Discov.*, vol. 24, no. 3, pp. 613-662, 2012. doi:http://dx.doi.org/10.1007/s10618-012-0253-2

[23] Z. Jeremić, J. Jovanović, and D. Gašević, "Personal learning environments on the Social Semantic Web," *Semantic Web - Linked Data for science and education*, vol. 4, no. 1, pp. 23-51, 2013. doi:http://dx.doi.org/10.3233/SW-2012-0058

[24] O. Khriyenko, and M. Nagy, "Semantic Web-driven Agent-based Ecosystem for Linked Data and Services." pp. 110-117, 2011

[25] F. Lécué, S. Tallevi-Diotallevi, J. Hayes, R. Tucker, V. Bicer, M. Sbodio, and P. Tommasi, "Smart traffic analytics in the semantic web with STAR-CITY: Scenarios, system and lessons learned in Dublin City," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 27–28, pp. 26-33, 2014. doi:http://dx.doi.org/10.1016/j.websem.2014.07.002

[26] L. D. Ngan, and R. Kanagasabai, "Semantic Web service discovery: state-of-the-art and research challenges," *Personal and Ubiquitous Computing*, vol. 17, no. 8, pp. 1741-1752, 2013. doi:http://dx.doi.org/10.1007/s00779-012-0609-z

[27] K. A. Demir, "Multi-View Software Architecture Design: Case Study of a Mission-Critical Defense System," *Computer and Information Science*, vol. 8, no. 4, pp. 12-31, 2015

[28] A. Aleti, B. Buhnova, L. Grunske, A. Koziolek, and I. Meedeniya, "Software Architecture Optimization Methods: A Systematic Literature Review," *IEEE Transactions on Software Engineering*, vol. 39, no. 5, pp. 658-683, 2013. doi:http://dx.doi.org/10.1109/TSE.2012.64

[29] E. Ginters, M. Schumann, A. Vishnyakov, and S. Orlov, "Software Architecture and Detailed Design Evaluation," *Procedia Computer Science*, vol. 43, pp. 41-52, 2015. doi:http://dx.doi.org/10.1016/j.procs.2014.12.007

[30] C. Yang, P. Liang, and P. Avgeriou, "A systematic mapping study on the combination of software architecture and agile development," *Journal of Systems and Software*, vol. 111, pp. 157-184, 2016. doi:http://dx.doi.org/10.1016/j.jss.2015.09.028

[31] M. Oussalah, F. Bhat, K. Challis, and T. Schnier, "A software architecture for Twitter collection, search and geolocation services," *Knowledge-Based Systems*, vol. 37, pp. 105-120, 2013. doi:http://dx.doi.org/10.1016/j.knosys.2012.07.017

[32] R. Capilla, A. Jansen, A. Tang, P. Avgeriou, and M. A. Babar, "10 years of software architecture knowledge management: Practice and future," *Journal of Systems and Software*, vol. 116, pp. 191-205, 2016. doi:http://dx.doi.org/10.1016/j.jss.2015.08.054

[33] A. R. d. M. Neves, A. M. G. Carvalho, and C. G. Ralha, "Agent-based architecture for context-aware and personalized event recommendation," *Expert Systems with Applications*, vol. 41, no. 2, pp. 563-573, 2014. doi:http://dx.doi.org/10.1016/j.eswa.2013.07.081

[34] R. Brisebois, A. Abran, and A. Nadembega, "A Semantic Metadata Enrichment Software Ecosystem (SMESE) based on a Multi-platform Metadata Model for Digital Libraries," *Journal of Software Engineering and Applications (JSEA)*, vol. 10, pp. 370-405, 2017. doi:http://dx.doi.org/10.4236/jsea.2017.104022

[35] G. H. Alférez, V. Pelechano, R. Mazo, C. Salinesi, and D. Diaz, "Dynamic adaptation of service compositions with variability models," *Journal of Systems and Software*, vol. 91, pp. 24-47, 2014. doi:http://dx.doi.org/10.1016/j.jss.2013.06.034

[36] K. Bontcheva, J. Kieniewicz, S. Andrews, and M. Wallis, "Semantic Enrichment and Search: A Case Study on Environmental Science Literature," *D-Lib Magazine*, vol. 21, no. 1-2, pp. 1-18, 2015

[37] R. Krueger, D. Thom, and T. Ertl, "Semantic Enrichment of Movement Behavior with Foursquare-A Visual Analytics Approach," *IEEE Transactions on Visualization and Computer Graphics*, vol. 21, no. 8, pp. 903-915, 2015. doi:http://dx.doi.org/10.1109/TVCG.2014.2371856

[38] C. Kunze, and R. Hecht, "Semantic enrichment of building data with volunteered geographic information to improve mappings of dwelling units and population," *Computers, Environment and Urban Systems*, vol. 53, pp. 4-18, 2015. doi:http://dx.doi.org/10.1016/j.compenvurbsys.2015.04.002

[39] R. Fileto, V. Bogorny, C. May, and D. Klein, "Semantic enrichment and analysis of movement data: probably it is just starting!," *SIGSPATIAL Special*, vol. 7, no. 1, pp. 11-18, 2015. doi:http://dx.doi.org/10.1145/2782759.2782763

[40] R. Fileto, C. May, C. Renso, N. Pelekis, D. Klein, and Y. Theodoridis, "The Baquara2 knowledge-based framework for semantic enrichment and analysis of movement data," *Data & Knowledge*

Engineering, vol. 98, pp. 104-122, 2015. doi:http://dx.doi.org/10.1016/j.datak.2015.07.010

[41] A. ANGJELI, "ISNI, For an worldwide Identification Ecosystem," *Institut National de l'Histoire de l'Art (INHA)*, 2016

Authors' Profiles

Ronald Brisebois is currently a PhD student at the École de Technologie Supérieure (ETS) – Université du Québec (Montréal, Canada). He received a B. Science in Physics at University of Montreal in 1983, a BA in Computer Science at University of Quebec in 1985 and his MBA at Hautes Etudes Commerciales - HEC (Business School) in 1989. From 1989 to 1993, **Ronald Brisebois** was a professor of Software Engineering at the University of Sherbrooke. His PhD research focus on semantic web, artificial intelligence, autonomous software architecture, new generation software designing, enriched metadata modeling and software engineering.

Renowned entrepreneur in the field of information technology, Ronald Brisebois has held management positions in various top-level firms (Caisses populaires Desjardins). In 1991, he was a professor at the University of Sherbrooke; in 1992, he founded his first company. Cognicase Inc. quickly became one of the largest players in the information technology field in Canada. In 2003, Ronald created Mondo/Stellar, one of the leading providers of integrated solutions for public libraries, academic institutions, specialized and consortia systems worldwide.

Dr. Abran holds a Ph.D. in Electrical and Computer Engineering (1994) from École Polytechnique de Montréal (Canada) and master degrees in Management Sciences (1974) and Electrical Engineering (1975) from University of Ottawa (Canada).

He is a professor at the École de Technologie Supérieure (ETS) – Université du Québec (Montréal, Canada). He has over 20 years of experience in teaching in a university environment as well as 20 years of industry experience in information systems development and software engineering management. His research interests include software productivity and estimation models, software engineering foundations, software quality, software functional size measurement, software risk management and software maintenance management. He has published over 400 peer-reviewed papers. He is the author of the books 'Software Project Estimation', 'Software Metrics and Software Metrology' and a co-author of the book 'Software Maintenance Management' (Wiley Interscience Ed. & IEEE-CS Press).

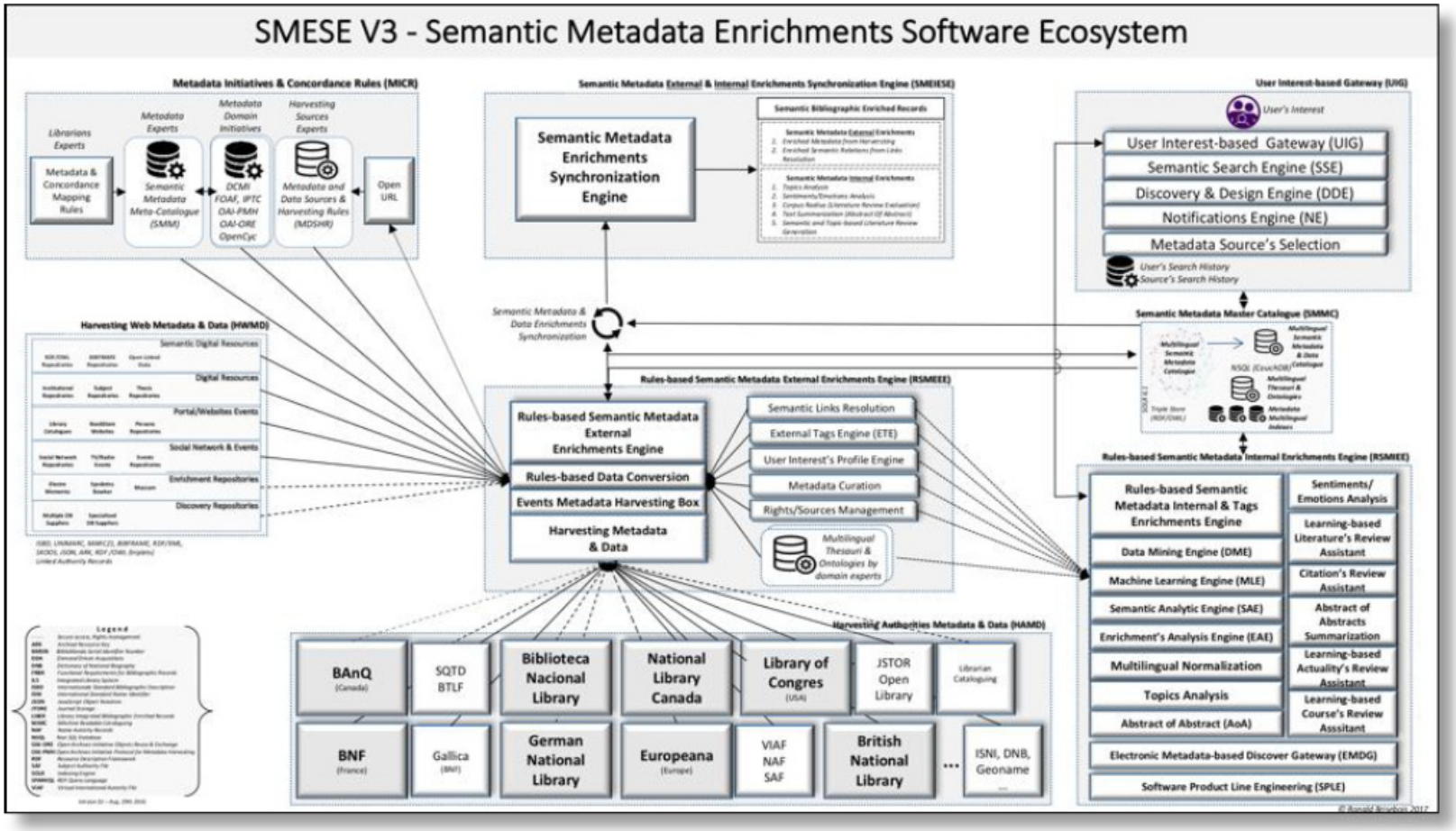
Dr. Abran is also the 2004 co-executive editor of the Guide to the Software Engineering Body of Knowledge – SWEBOK (see ISO 19759 and www.swebok.org) and he is the chairman of the Common Software Measurement International Consortium (COSMIC) – http://cosmic-sizing.org/. A number of Dr. Abran research works have influenced international standards in software engineering (i.e., ISO 19761, ISO 19759, ISO 14143-3, etc.).

Dr. Apollinaire Nadembega is currently a guest member of the Network Research Laboratory (NRL) of the University of Montreal. He received his B. E degree in Information Engineering from Computer Science High School, Bobo-Dioulasso, Burkina faso in 2003, his Master's degree in computer science from the Arts and Business Institute, Ouagadougou, Burkina faso in 2007 and his Ph.D. degree in mobile networks from the University of Montreal, Montreal, QC, Canada in 2014. The primary focus of his Ph.D. thesis is to propose a mobility model and bandwidth reservation scheme that supports quality-of-service management for wireless cellular networks. Dr. Nadembega's research interests lie in the field of artificial intelligence, machine learning, networking modelling, semantic web, metadata management system, software architecture, mobile multimedia streaming, call admission control, bandwidth management and mobile cloud computing.

From 2004 to 2008, he was a programming engineer with Burkina Faso public administration staff management office.

Manuscript received Month Date, Year; revised Month Date, Year; accepted Month Date, Year.

APPENDIX A: FIG. 3. - SMESE FRAMEWORK: SEMANTIC ENRICHED METADATA SOFTWARE ECOSYSTEM



APPENDIX B: FIG. 7. - SMESE METADATA MODEL

Id.	Unimarc	Marc21	propertyName(Excel)	Description	BM label Fr	BM label En	BM label Sp	BM label Ne	BM label Gr	MetaData type	Expected response	isUrl	Repeatable	isRelated	relatedContentType	nestedRelation	Thesaurus	Enriched	isTranslatable	Data Location	Mixin	Hidden (Form level)	isNotDisplayable (bibliographic record level)	Short Notice	Onglet Edit	SoureeOfSchema	InMedia Version
M1-m	200$acdehi	130$anplskfdmor 245$ahbcnp 247$abfgnp 700$t	title	It is the main title of a content	Titre	Title	Titulo				Multilingue											X	X		Desc	BM	
M2-m	7XX	100$aqbcdtnplskfgu 110$aqbcdtnplskfgu 111$aqbcdtnplskfgu 971$a4 3100$a4	author	content. Please note that author is special in that HTML 5 provides a special mechanism for indicating authorship via the rel tag. That is equivalent to this and	Auteur	Author	Autor				Text		X	X	domain:personality		Personnes ou Organisations			Local				X	Desc	BNF	
M3-m	700$abcdfg4 701$abcdfg4 710$abcdefgh4 711$abcdefgh4 720$af4 721$af4	100$aqbcdtnplskfgu 110$aqbcdtnplskfgu 111$aqbcdtnplskfgu 971$a4 3100$a4	creator	The creator/author of this CreativeWork. This is the same as the Author property for CreativeWork	Auteur	Author	Autor				Text		X	X	domain:personality		Personnes ou Organisations			Local				X	Desc	BNF	
M4-m	7xx	1XX	promoter	du contenu; c'est le	Organisateur	Promoter	Promotor				Text		X	X	sonality		nes			Local					Desc	BM	
M5-m	206$abcdef 230$a 215$acde	255$abcdefg 256$a 300$abceg	description	C'est une texte qui décrit le contenu	Description	Description	Descripción				Multilingue		X						X	Local		X	X		Desc	BNF	
M6-m			Image	An image of the item. This can be a image file or a URL or a fully described ImageObject	Imagette	Thumbnail					Image		X					X		BM.com					Desc	BNF	

APPENDIX C: FIG. 8. - EXAMPLE OF A SMESE SEMANTIC MATRIX MODEL

Id.	Unimarc	Marc21	propertyName(Excel)	Poster	CartographicMat	Serial (Périodique)	Book	Physical	Image	Event	BookChapter	PressRelease	Law (loi et règlements)	Research Report (Rapport de recherche)	MDAdministratives	Organization Report (Rapport d'organisme)	Thesis	ConferencePaper	ConferenceProceedings	GeneralAssembly	Speech (Discours/Allocution)	Periodicalissue (Numéro de périodique)	Periodical Article (Article de Périodique)
M1-m	200$acdehi	130$anplskfdmor 245$ahbcnp 247$abfgnp 700$t	title	3	3	13	13	13	6	3	4	6	3	41		30	3	35	35	1	1	3	9
M2-m	7XX	100$aqbcdtnplskfgu 110$aqbcdtnplskfgu 111$aqbcdtnplskfgu 971$a4 3100$a4	author	4	4	126	241	134	72		3	3	4				4	42	42	2	2		4
M3-m	700$abcdfg4 701$abcdfg4 710$abcdefgh4 711$abcdefgh4 720$af4 721$af4	100$aqbcdtnplskfgu 110$aqbcdtnplskfgu 111$aqbcdtnplskfgu 971$a4 3100$a4	creator			3	3	5	3		48	69		72		58	95	89	89	3	3		100
M4-m	7xx	1XX	promoter							47									90	4	4		
M5-m	206$abcdef 230$a 215$acde	255$abcdefg 256$a 300$abceg	description	15	15	39	35	50	7	13	41	88	5	114		21	5	127	132	62	57	4	134
M6-m			Image			84	71	80	70	15	42												

Paper 3:

A Semantic Metadata Enrichment Software Ecosystem based on Sentiment and Emotion Metadata Enrichments

Ronald Brisebois, Alain Abran, Apollinaire Nadembega, Philippe N'techobo

http://ijsrset.com/PDF.php?pid=2466&v=3&i=2&y=2017&m=March-April

© 2017 IJSRSET | Volume 3 | Issue 2 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

A Semantic Metadata Enrichment Software Ecosystem based on Sentiment and Emotion Metadata Enrichments

Ronald Brisebois[1], Alain Abran[1], Apollinaire Nadembega*[2], Philippe N'techobo[3]

[1]École de technologie supérieure, University of Quebec, Montreal, Quebec, Canada

*[2]Network Research Lab., University of Montreal, Montreal, Quebec, Canada

[3]École Polytechnique de Montréal, Montreal, Quebec, Canada

ABSTRACT

Information retrieval and analysis is frequently used to extract meaningful knowledge from the unstructured web and long texts. As existing computer search engines struggle to understand the meaning of natural language, semantically sentiment and emotion enriched metadata may improve search engine capabilities and user finding. A semantic metadata enrichment software ecosystem (SMESE) has been proposed in our previous research. This paper presents an enhanced version of this ecosystem with a sentiment and emotion metadata enrichments algorithm. This paper proposes a model and an algorithm enhancing search engines finding contents according to the user interests, through text analysis approaches for sentiment and emotion analysis. It presents the design, implementation and evaluation of an engine harvesting and enriching metadata related to sentiment and emotion analysis. It includes the SSEA (Semantic Sentiment and Emotion Analysis) semantic model and algorithm that discover and enrich sentiment and emotion metadata hidden within the text or linked to multimedia structure. The performance of sentiment and emotion analysis enrichments is evaluated using a number of prototype simulations by comparing them to existing enriched metadata techniques. The results show that the algorithm SSEA enable greater understanding and finding of document or contents associated with sentiment and emotion enriched metadata.

Keywords: Emotion Analysis, Natural Language Processing, Semantic Metadata Enrichment, Sentiment Analysis, Text And Data Mining

I. INTRODUCTION

Semantic information retrieval (SIR) is the science of searching semantically for information within databases, documents, texts, multimedia files, catalogues and the web. The human brain has an inherent ability to detect sentiment and emotion in written or spoken language. However, the internet, social media and repositories have expanded the number of sources, volume of information and number of relationships so fast that it has become difficult to process all this information [1].
Finding bibliographic references or semantic relationships in texts makes it possible to localize specific text segments using ontologies to enrich a set of semantic metadata related to sentiment or emotion. This paper presents an enhanced SMESE model and prototype [2] using metadata from linked open data, structured data, metadata initiatives, concordance rules and authorities metadata.

The current methodology proposed by SIR researchers for text analysis within the context of entity metadata enrichment (EME) reduces each document in the corpus to a vector of real numbers where each vector represents ratios of counts. Several EME approaches have been proposed, most of them making use of term frequency–inverse document frequency (tf-idf) [3, 4]. In the tf-idf scheme, a basic vocabulary of "words" or "terms" is chosen, then for each document in the corpus, a frequency count is calculated from the number of occurrences of each word [3, 4]. After suitable normalization, the frequency count is compared to an inverse document frequency count (e.g the inverse of the number of documents in the entire corpus where a

given word occurs — generally on a log scale, and again suitably normalized). The end result is a term-by-document matrix X whose columns contain the tf-idf values for each of the documents in the corpus. Thus the tf-idf scheme reduces documents of arbitrary length to fixed-length lists of numbers. For non-textual content, tools are available to extract the text from multimedia entities. For example, Bougiatiotis and Giannakopoulos [5] propose an approach that extracts topical representations of movies based on mining of subtitles. This paper focuses on contributions to mainly one EME research fields: sentiment analysis (SA) including emotion analysis.

The main objective of SA is to establish the attitude of a given person with regard to sentences, paragraphs, chapters or documents [1, 4, 6-12]. Indeed, many websites offer reviews of items like books, cars, mobiles, movies etc., where products are described in some detail and evaluated as good/bad, preferred/not preferred; unfortunately, these evaluations are insufficient for users in order to help them to make decision. In addition, with the rapid spread of social media, it has become necessary to categorize these reviews in an automated way [4]. For this automatic classification, there are different methods to perform SA, such as keyword spotting, lexical affinity and statistical methods. However, the most commonly applied techniques to address the SA problem belong either to the category of text classification supervised machine learning, which uses methods like naïve Bayes, maximum entropy or support vector machine (SVM), or to the category of text classification unsupervised machine learning (UML). Also, fuzzy sets appear to be well-equipped to model sentiment-related problems given their mathematical properties and ability to deal with vagueness and uncertainty — characteristics that are present in natural languages processing.

Thus, a combination of techniques may be successful in addressing SA challenges by exploiting the best of each technique. In addition, the semantic web may be a good solution for searching relevant information from a huge repository of unstructured web data [6].

According to [7], the SA process typically consists of a series of steps:

1. Corpus or data acquisition
2. Text preprocessing
3. Opinion mining core process
4. Aggregation and summarization of results
5. Visualization

One current limitation in the area of SA research is its focus on sentiment classification while ignoring the detection of emotions. For example, document emotion analysis may help to determine an emotional barometer and give the reader a clear indication of excitement, fear, anxiety, irritability, depression, anger and other such emotions. For this reason, our research focuses on sentiment and emotion analysis (SEA) instead of SA.

A number of algorithms are used to perform text mining, including: latent Dirichlet allocation (LDA) [13], tf-idf [3, 4], latent semantic analysis (LSA) [14], formal concept analysis (FCA) [15], latent tree model (LTM) [16], naïve Bayes (NB) [17], support vector machine method (SVM) [17], artificial neural network (ANN) [18] based on the associated document's features.

Our approach improves the accuracy of sentiment and emotion discovery by semantically enriching the metadata from the linked open data and the bibliographic records. This paper presents the design, implementation and evaluation of an enhanced ecosystem, called semantic metadata enrichment ecosystem or SMESE. It includes:

1. An enhanced semantic metadata catalogue.
2. An enhanced harvesting of metadata & data engine.
3. Metadata enrichment based on semantic topic detection and sentiment/emotion analysis.

More specifically, this paper extends our previous work [2] with:

1. SSEA: discovery of sentiments/emotions hidden within the text or linked to a multimedia structure through an AI computational approach.
2. Algorithm for generation of semantic topics by text analysis, relationships and multimedia contents; this second algorithm will be proposed in another paper.

Using simulation, the performance of SSEA was evaluated in terms of accuracy of sentiment and emotion discovery. Existing approaches to enriching metadata, in terms of sentiment and emotion discovery were used for comparison. Simulation results showed that SSEA outperforms existing approaches.

The remainder of the paper is organized as follows. Section 2 presents the related work. Section 3 describes SSEA algorithm. Section 4 presents the evaluation through a number of simulations while Section 5 presents a summary and some suggestions for future work.

II. RELATED WOK

In the past few years, a number of natural language processing (NLP) tasks have been configured for semantic web (SW) tasks including: ontology learning, linked open data, entity resolution, natural language querying to linked data, etc. [19]. This improvement of metadata enrichment using SW involves obtaining hidden data, hence the concept of entity metadata extraction (EME).

Interest in EME was initially limited to those in the SW community who preferred to concentrate on manual design of ontologies as a measure of quality. Following linked data bootstrapping provided by DBpedia, many changes ensued with a consequent need for substantial population of knowledge bases, schema induction from data, natural language access to structured data, and in general all applications that make for joint exploitation of structured and unstructured content. In practice, NLP research started using SW resources as background knowledge. Graph-based methods, meanwhile, were incrementally entering the toolbox of semantic technologies at large.

In the related work section, sentiment and emotion analysis (SEA) that is one field of entity metadata extraction research from text aspect is investigated.

A. Sentiment analysis

The problem of sentiment analysis has been widely studied and different approaches applied, such as machine learning (ML), natural language processing (NLP) and semantic information retrieval (SIR).

There are three main techniques for sentiment analysis [20]:

1. Keyword spotting.
2. Lexical affinity.
3. Statistical methods.

Keyword spotting includes developing a list of keywords that relate to a certain sentiment. These words are usually positive or negative adjectives since such words can be strong indicators of sentiment. Keyword spotting classifies text by affect categories based on the presence of unambiguous affect words such as happy, sad, afraid, and bored.

Lexical affinity is slightly more sophisticated than keyword spotting. Rather than simply detecting obvious affect words, it assigns to arbitrary words a probabilistic 'affinity' for a particular emotion. Lexical affinity determines the polarity of each word using different unsupervised techniques. Next it aggregates the word scores to obtain the polarity score of the text. For example, 'accident' might be assigned a 75% probability of indicating a negative effect, as in 'car accident' or 'injured in an accident'.

Statistical methods, such as Bayesian inference and support vector machines, are supervised approaches in which a labeled corpus is used for training a classification method which builds a classification model used for predicting the polarity of novel texts. By feeding a large training corpus of affectively annotated texts to a machine learning algorithm, it is possible for the system to not only learn the affective valence of affect keywords (as in the keyword spotting approach), but also to take into account the valence of other arbitrary keywords (like lexical affinity), punctuation, and word co-occurrence frequencies. In addition, sophisticated NLP techniques have been developed to address the problems of syntax, negation and irony. Sentiment analysis can be carried out at different levels of text granularity: document [17, 21-25], sentence [1, 4, 6, 26, 27], phrase [28], clause, and word [18, 29, 30].

Sentiment analysis may be at the sentence or phrase level (which has recently received quite a bit of research attention) or at the document level.

From the perspective of this paper, our work may be seen as document-level sentiment analysis—that is, a document is regarded as an opinion on an entity or aspect of it. This level is associated with the task called document-level sentiment classification, i.e., determining whether a document expresses a positive or negative sentiment.

In [8], the authors presented a survey of over one hundred articles published in the last decade on the tasks, approaches, and applications of sentiment analysis. With a major part of available worldwide data being unstructured (such as text, speech, audio, and video), this poses important research challenges. In recent years numerous research efforts have led to automated SEA, an extension of the NLP area of research. The authors identified seven broad classifications:

1. Subjectivity classification
2. Sentiment classification
3. Review usefulness measurement
4. Lexicon creation
5. Opinion word and product aspect extraction
6. Opinion spam detection
7. Various applications of opinion mining

The first five dimensions represent tasks to be performed in the broad area of SEA. For the first three dimensions (subjectivity classification, sentiment classification and review usefulness measurement), the authors note that the applied approaches are broadly classified into three categories:

1. Machine learning
2. Lexicon based
3. Hybrid approaches

Since one of our research objectives was to extract sentiment and emotion metadata from documents, the rest of this section focuses on sentiment classification, lexicon creation, and opinion word and product aspect extraction. Sentiment classification is concerned with determining the polarity of a sentence; that is, whether a sentence is expressing positive, negative or neutral sentiment towards the subject. A lexicon is a vocabulary of sentiment words with respective sentiment polarity and strength value while opinion word and product aspect extraction is used to identify opinion on various parts of a product. As per our research objective the rest of the literature review was oriented to document-level sentiment analysis. For our purposes, we assume that a document expresses sentiments on a single content and is written by a single author.

Cho et al. [23] proposed a method to improve the positive vs. negative classification performance of product reviews by merging, removing, and switching the entry words of the multiple sentiment dictionaries. They merge and revise the entry words of the multiple sentiment lexicons using labeled product reviews. Specifically, they selectively remove the sentiment words from the existing lexicon to prevent erroneous matching of the sentiment words during lexicon-based sentiment classification. Next, they selectively switch the polarity of the sentiment words to adjust the sentiment values to a specific domain. The remove and switch operations are performed using the target domain's labeled data, i.e. online product reviews, by comparing the positive and negative distribution of the labeled reviews with a positive and negative distribution of the sentiment words. They achieved 81.8% accuracy for book reviews. However, their contribution is limited to development of a novel method of removing and switching the content of the existing sentiment lexicons. Moraes et al. [17] compared popular machine learning approaches (SVM and NB) with an ANN-based method for document-level sentiment classification. Naive Bayes (NB) is a probabilistic learning method that assumes terms occur independently while the support vector machine method (SVM) seeks to maximize the distance to the closest training point from either class in order to achieve better generalization/classification performance on test data. The authors reported that, despite the low computational cost of the NB technique, it was not competitive in terms of classification accuracy when compared to SVM. According to the authors, many researchers have reported that SVM is perhaps the most accurate method for text classification. Artificial neural network (ANN) derives features from linear combinations of the input data and then models the output as a nonlinear function of these features. Experimental results showed that, for book datasets, SVM outperformed ANN when the number of terms exceeded 3,000. Although SVM required less training time, it needed more running time than ANN. For 3,000 terms, ANN required 15 sec training time (with negligible running time) while SVM training time was negligible (1.75 sec). In addition, their contribution was limited to performing comparisons between existing approaches. As in [17], Poria S. et al. [31] experimented with existing approaches and showed that SVM is a better approach for text-based emotion detection.

B. Emotion analysis

This section focuses on sentiment and emotion analysis. Emotions include the interpretation, perception and response to feelings related to the experience of any

particular situation. Emotions are also associated with mood, temperament, personality, outlook and motivation [20, 32, 33]; indeed, the concepts of emotion and sentiment have often been used interchangeably, mostly because both refer to experiences that result from combined biological, cognitive, and social influences. However, sentiments are differentiated from emotions by the duration in which they are experienced. Emotions are brief episodes of brain, autonomic, and behavioral changes. Sentiments have been found to form and be held over a longer period and to be more stable and dispositional than emotions. Moreover, sentiments are formed and directed toward an object, whereas emotions are not always targeted toward an object.

The emotion-topic model (ETM) [34], SWAT model and emotion-term model (ET) [34] are the state-of-the-art models. The SWAT model was proposed to explore the connection between the evoked emotions of readers and news headlines by generating a word-emotion mapping dictionary. For each word w in the corpus, it assigns a weight for each emotion e, i.e., $P(e|w)$ is the averaged emotion score observed in each news headline H in which w appears. The emotion-term model is a variant of the NB classifier and was designed to model word-emotion associations. In this model, the probability of word wj conditioned on emotion ek is estimated based on the co-occurrence count between word wj and emotion ek for all documents. The emotion-topic model is combination of the emotion-term model and LDA. In this model, the probability of word wj conditioned on emotion ek is estimated based on the probability of latent topic z conditioned on emotion ek and the probability of word wj conditioned on latent topic z.

A number of techniques exist to detect emotions [35]:

1. *Audio based emotion detection*: information from the spectral elements in voice (e.g., speaking rate, pitch, energy of speech, intensity, rhythm regularity, tempo and stress distribution) is used to gather clues about emotions. The features extracted are compared with the training sets in the database using the classifiers.
2. *Blue eyes technology* based on eye moment. In this technique, a picture of the person whose emotions are to be detected is taken and the portion showing his or her eyes is extracted. This extracted image is converted from RGB form to a binary image and compared with ideal eye images depicting various emotions stored in the database. Once the match between the extracted image and one in the database is found, the type of emotion (i.e. happiness, anger, sadness or surprise) is said to be detected.
3. *Facial expression based emotion detection* based on photos of the individual. The images are processed for skin segmentation and analyzed as follows. The image is contrasted, separating the brightest and darkest color in the image area and discriminating the pixels between skin and non-skin. The image is converted into binary form. This processed image is then compared with images forming the training sets in classifiers.
4. *Handwriting based emotion detection* is based on various handwriting indicators or traits of writing (e.g., baseline, slant, pen-pressure, size, zone, strokes, spacing, margins, loops, 'i'-dots, 't'-bar, etc.).
5. *Text based emotion detection* where a computerized NLP approach is used to analyze written text to detect the emotions of the writer. The document is first preprocessed by normalizing the text, then keywords indicating emotional features are extracted. Corresponding emotions are identified through various approaches such as:

a) Keyword spotting technique.
b) Lexical affinity method.
c) Learning based methods.
d) Hybrid method, or by using an emotion ontology which stores a range of emotion classes, associated keywords and relationships.

Text-based emotion detection approaches focus on 'optimistic', 'depressed' and 'irritated.' The limitations are:

1. Ambiguity of keyword definitions.
2. Inability to recognize sentences without keyword.
3. Difficulty determining emotion indicators.

Lei et al. [36] adopted the lexicon-based approach in building the social emotion detection system for online news based on modules of document selection, part-of-speech (POS) tagging, and social emotion lexicon generation. First, they constructed a lexicon in which each word is scored according to multiple emotion labels such as joy, anger, fear, surprise, etc. Next, a

lexicon was used to detect social emotions of news headlines. Specifically, given the training set T and its feature set F, an emotion lexicon is generated as a V×E matrix where the (*j*, *k*) item in the matrix is the score (probability) of emotion *ek* conditioned on feature *fj*. The authors do not explain how they extracted the features from the document.

Anusha and Sandhya [37] proposed a system for text-based emotion detection which uses a combination of machine learning and natural language processing techniques to recognize affect in the form of six basic emotions proposed by Ekman. They used the Stanford CoreNLP toolkit to create the dependency tree based on word relationships. Next, phrase selection is done using the rules on dependency relationships that gives priority to the semantic information for the classification of a sentence's emotion. Based on the phrase selection, they used the Porter stemming algorithm for stemming, and stopwords removal and tf-idf to build the feature vectors. The authors do not propose a new approach but implement existing algorithms.

Cambria et al. [38] explored how the high generalization performance, low computational complexity, and fast learning speed of extreme learning machines can be exploited to perform analogical reasoning in a vector space model of affective common-sense knowledge. After performing TSVD on AffectNet, they used the Frobenius norm to derive a new matrix. For the emotion categorization model, they used the Duchenne smile and the Klaus Scherer model. As in [37], the authors do not propose a new approach but implement existing algorithms.

III. RESULTS AND DISCUSSION

Table I: Summary of attribute comparison of existing and SSEA algorithm

Existing algorithms	extraction	Classification	Sentiment analysis	Emotion analysis	Concept extraction
AlchemyAPI (http://www.alchemyapi.com/)	x	x	x	x	x
DBpedia Spotlight (https://github.com/dbpedia-spotlight)					x
Wikimeta (https://www.w3.org/2001/sw/wiki/Wikimeta)					x
Yahoo! Content Analysis API (https://developer.yahoo.com/contentanalysis/)		x			x
Open Calais (http://www.opencalais.com/)	x	x			x
Tone Analyzer (https://tone-analyzer-demo.mybluemix.net/)			x	x	
Zemanta (http://www.zemanta.com/)					x
Receptiviti (http://www.receptiviti.ai/)			x	x	
Apache Stanbol (https://stanbol.apache.org/)					x
Bitext (https://www.bitext.com/)			x		x
Mood patrol (https://market.mashape.com/soulhackerslabs/moodpatrol-emotion-detection-from-text)				x	
Aylien (http://aylien.com/)	x	x	x		
AIDA (http://senseable.mit.edu/aida/)					x
Wikifier (http://wikifier.org/)					x
TextRazor (https://www.textrazor.com/)					x
Synesketch (http://krcadinac.com/synesketch/)				x	
Toneapi (http://toneapi.com/)			x	x	
SSEA algorithm	x	x	x	x	x

1. Rule-Based Semantic Metadata Internal Enrichment Engine

This section presents an overview and details of the proposed rule-based semantic metadata internal enrichment engine, including the SSEA algorithm used to process semantic metadata internal enrichment. The main goal of this paper is to enhance the SMESE platform [2] through text analysis approaches for sentiment and emotion and detection.

C. Rule-based semantic metadata internal enrichment engine overview

The rule-based semantic metadata internal enrichment engine has been designed to find short descriptions, in terms of topics, sentiments and emotions of the members of a collection to enable efficient processing of large collections while preserving the semantic and statistical relationships that are useful for tasks such as: topic detection, classification, novelty detection, summarization, and similarity and relevance judgments. Figure 1 shows an overview of the architecture that consists of:

1. User interest-based gateway.
2. Metadata initiatives & concordance rules.
3. Harvesting web metadata & data.
4. User profiling engine.
5. Rule-based semantic metadata internal enrichment engine.

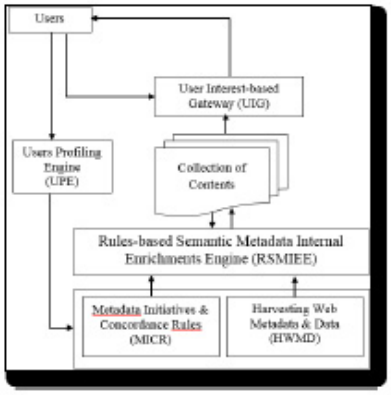


Figure 1: Architecture of the rule-based semantic metadata internal enrichment engine

The user interest-based gateway (UIG) is designed to push notifications to users based on the emotions and interests found using the user-profiling engine. UIG is also a discovery tool that allows users to search and discover contents based on their interests and emotions.
The user-profiling engine applies machine learning algorithms to user feedback in terms of appreciation, rating, comment and historical research in order to provide user profiles. When the contextual information of users is available, it is used to increase the accuracy of the profiling process.

The engine performs automated metadata internal enrichment based on the set of metadata initiatives & concordance rules, the engine for harvesting web metadata & data, the user profile and a thesaurus. This engine implements SSEA for sentiment and emotion detection of documents and an algorithm for topic-automated detection from documents.

SSEA tasks may be redefined as document classification issues as they contain methods for the classification of natural language text. These methods will help to predict the query's category, given a set of training documents with known categories and a new document, which is usually called the query.
The following sub-sections present the terminology and assumptions, the necessary pre-processing and details of the algorithms implemented in the engine.

D. Terminology and assumptions

In this section the following terms are defined:

1. A word or term is the basic unit of discrete data, defined to be an item from a vocabulary indexed by {1, …,V}. Terms are presented using unit-basis vectors that have a single component equal to one and all other components equal to zero. Thus, using superscripts to denote components, the ith term in the vocabulary is represented by an I-vector *w* such that $w^i = 1$ and $w^j = 0$ for $i \neq j$. For example, let V= {book, image, video, cat, dog} be the vocabulary. The video term is represented by the vector (0, 0, 1, 0, 0).
2. A line is a sequence of N terms denoted by *l*. These terms are extracted from a real sentence; a sentence is a group of words, usually containing a verb, that expresses a thought in the form of a statement, question, instruction, or exclamation and when written begins with a capital letter.
3. A document is a sequence of N lines denoted by D = $(w_1,w_2; \ldots,w_N)$, where w_i is the ith term in the sequence coming from the lines. D is represented by its lines as $D = (l_1, \ldots l_i,\ldots,l_K)$.
4. A corpus is a collection of *M* documents denoted by $C = \{D_1, D_2, \ldots, D_M\}$.
5. An emotion word is a word with strong emotional tendency. An emotion word is a probabilistic distribution of emotions and represents a semantically coherent emotion analysis. For example, the word "excitement", presenting a positive and pleased feeling, is assigned a high probability to emotion "joy".

To implement the SSEA algorithm, an initial set of conditions must be established:

1. A list of topics $T = \{t_1, \ldots , t_i, \ldots , t_n\}$ is readily available.
2. Each existing document D_j is already annotated by topic. The annotated topics of document D_j are denoted as $T_{Dj} = \{t_p \ldots, t_i , \ldots, t_q\}$ where t_p, t_i, and $t_q \in T$.
3. The corpus of documents is already classified by topics. $C_n=\{\ldots,D_j,\ldots\}$ denotes the corpus of documents that have been annotated with topic t_i. Note that the document D_j may be located in several corpuses.
4. A list of emotions $E = \{e_1, \ldots , e_i, \ldots , e_E\}$ is readily available with the common instances of *e* being joy, anger, fear, surprise, touching, empathy, boredom, sadness, warmth.
5. A set of ratings over E emotion labels denoted by $R_{Dj} = \{r_{d,e1} \ldots, r_{d,ei} , \ldots, r_{d,eE}\}$. The value of $r_{d,ei}$ is the number of users who have voted i^{th} emotion label e_i for document *d*. In other words, $r_{d,ei}$ is the number of

users who claimed that emotion e_i is found in document d.

6. The corpus of documents are already classified by sentiment and emotion based on the user rating. $C_{ei} = \{\ldots,D_j,\ldots\}$ denotes the corpus of documents rated with emotion e_i. Note that the document D_j may be located in several knowledge corpi.
7. A list of sentiments $S = \{s_1, \ldots, s_i, \ldots, s_S\}$ is readily available.
8. A thesaurus is available and has a tree hierarchical structure. A thesaurus contains a list of words with synonyms and related concepts. This approach uses synonyms or glosses of lexical resources in order to determine the emotion or polarity of words, sentences and documents.

E. Document Pre-Processing

Before document analysis, SSEA performs a pre-processing. The objective of the pre-processing is to filter noise and adjust the data format to be suitable for the analysis phases. It consists of stemming, phase extraction, part-of-speech filtering and removal of stop-words. The corpus of documents crawled from specific databases or the internet consists of many documents. The documents are pre-processed into a basket dataset C, called document collection. C consists of lines representing the sentences of the documents. Each line consists of terms, i.e. words or phrases. An example of C follows:

C=

……

……

D_j= (line 1) : term 1, term 2, term 6, term 9.

(line 2) : term 10, term 6, term 2, term 3.

……

(line j) : term 3, term 5, term 2, term 3, term 9, term 1.

……

(line N_j) : term 2, term 15, term 9, term 3, term 4.

……

……

More specifically, to obtain D_j, the following preprocessing steps are performed:

1. Language detection.
2. Segmentation: a process of dividing a given document into sentences.
3. Stop word: a process to remove the stop words from the text. Stop words are frequently occurring words such as 'a' an', the' that provide less meaning and generate noise. Stop words are predefined and stored in an array.
4. Tokenization: separates the input text into separate tokens.
5. Punctuation marks: identifies and treats the spaces and word terminators as the word breaking characters.
6. Word stemming: converts each word into its root form by removing its prefix and suffix for comparison with other words.

More specifically, a standard preprocessing such as tokenization, lowercasing and stemming of all the terms using the Porter stemmer [39]. Therefore, we also parse the texts using the Stanford parser [40] that is a lexicalized probabilistic parser which provides various information such as the syntactic structure of text segments, dependencies and POS tags. 'Word' and 'term' are used interchangeably in the rest of this paper.

F. Semantic sentiment and emotion analysis: SSEA

The aim of SSEA is to classify the corpus of documents taking emotion into consideration, and to determine which sentiment it more likely belongs to.

A document can be a distribution of emotion $p(e \mid d) e \in E$ and a distribution of sentiment $p(s \mid d) s \in S$. SSEA is a hybrid approach that combines a keyword-based approach and a rule-based approach. SSEA is applied at the basic word level and requires an emotional keyword dictionary that has keywords (emotion words) with corresponding emotion labels.
Next, to refine the detection, SSEA develops various rules to identify emotion. Rules are defined using an affective lexicon that contains a list of lexemes annotated with their affect.

The emotional keyword dictionary and the affective lexicon are implemented in a thesaurus. SSEA is a knowledge-based approach that uses an AI computational technique. The purpose of SSEA is to identify positive and negative opinions and emotions. Figure 2 presents an overview of the architecture of the sentiment and emotion detection process phase.

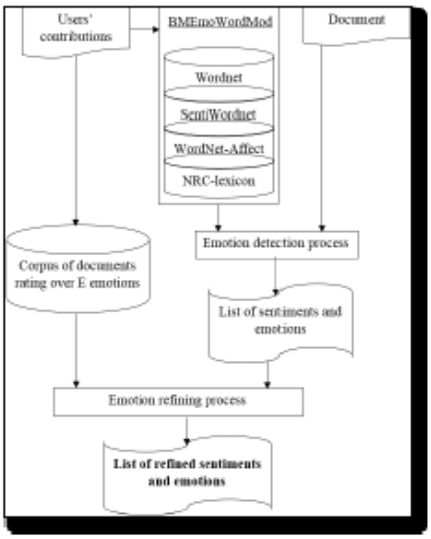


Figure 2: Sentiment and emotion detection process phase – Architecture overview

For affective text evaluation, SSEA uses the SS-Tagger (a part-of-speech tagger) [41] and the Stanford parser [40]. The Stanford parser was selected because it is more tolerant of constructions that are not grammatically correct. This is useful for short sentences such as titles. SSEA also uses several lexical resources that create the SSEA knowledge base located in the thesaurus. The lexical resources used are:

1. WordNet.
2. WordNet-Affect.
3. SentiWordNet.
4. NRC emotion lexicon.

WordNet is a semantic lexicon where words are grouped into sets of synonyms, called synsets. In addition, various semantic relations exist between these synsets (for example: hypernymy and hyponymy, antonymy and derivation). WordNet-Affect is a hierarchy of affective domain labels that can further annotate the synsets representing affective concepts. SentiWordNet assigns to each synset of WordNet three sentiment scores: positivity, negativity, objectivity, the sum of which always equals 1.0.

The NRC emotion lexicon is a list of English words and their association with eight basic emotions (anger, anticipation, disgust, fear, joy, sadness, surprise and trust) and two sentiments (negative and positive). The NRC emotion lexicon is a thesaurus that associates for a word, the value one or zero for each emotion. This association is made of binary vectors. The disadvantage of this thesaurus is that since the values are binary, all words belonging to an emotion have the same weight for that emotion. To address this problem, the NRC emotion lexicon thesaurus was combined with the WordNet, WordNet-Affect and SentiWordNet thesaurus. This associates a feelings score with each word-POS. POS1 are grammatical categories used to classify words in dimensions such as adjectives or verbs. SentiWordNet associates with each couple a valence score that can be either negative or positive with respect to the sense of the word in question. The word death, for example, is likely to have a negative score. SSEA also relies on shifter valences. These are lexical expressions capable of changing the valence score of emotions in a text.

For example, take the phrase "I am happy" with a score of 1 for the joy emotion. For the phrase "I am **very** happy", 'very' is a valence intensifier that will change the joy emotion score to 2. In the case, "I am **not** happy" the modifier 'not' will change the emotion joy to the contrary emotion sadness.

The main component of SSEA is the thesaurus, called BM emotion word model (BMEmoWordMod). BMEmoWordMod is an emotion-topic model that provides the emotional score of each keyword by taking the topic into account.

BMEmoWordMod introduces an additional layer (i.e., latent topic) into the emotion-term model such as SentiWordNet. SSEA is composed of three phases:

1. BMEmoWordMod generation process phase.
2. Sentiment and emotion discovery process phase.
3. Sentiment and emotion refining process phase.

The following sub-sections describe the three phases of the SSEA model used to discover sentiment and emotion.

1) BMEmoWordMod generation - process phase

In the first step, a training set from the original corpus is created. The most relevant and discriminative documents are selected automatically. In the second step, each word is tagged with a POS and the combination of word and POS used as the essential feature. Finally,

BMEmoWordMod is generated using the extracted features, which can then be used to discovery the sentiments and emotions of new documents.
Basically, a BMEmoWordMod entry has the following fields:

<Word/POS/synsets_ID><Topics><Emotion_Probability><Sentiment_Probability> where:

1. Emotion_Probability is a vector of ordered emotion label probability such as <anger probability, disgust probability, fear probability, joy probability, sadness probability, surprise probability>.
2. Sentiment_ Probability is a vector of ordered sentiment category probability such as <positive score, negative score>.

For example, the BMEmoWordMod entry for "kill" may look like: <kill/v/00829041><War><0.5, 0.1, 0.3, 0, 0.2, 0><0.1, 0.6>.

Step 1: Training set selection

The objective of this step is to reduce the time for generating the emotion lexicon BMEmoWordMod, while obtaining a better quality lexicon. For each emotion e_i, documents in the corpus are ranked by descending order of ratings over e_i. Next, the emotions with the highest ratings among the documents are chosen. Then relevant documents for a given emotion e_i are selected based on the topic detection algorithm; we assume that this topic detection algorithm is known. The training set selection process terminates when the first phase topic detection algorithm requirements are meet. The training set TS is produced by conducting this step on the entire corpus.

Step 2: Intermediate lexicon generation

Using WordNet-Affect, the WordNet entries are filtered in order to retain only those synsets where the A_labeb is "EMOTION". Then, using SentiWordNet and the NRC emotion lexicon, the sentiment category and emotion value are associated with each selected emotional synset of WordNet. An intermediate lexicon is produced where each entry is <word/POS/synsets_ID><Emotion_value><Sentiment_Score>.
BMEmoWordMod evaluates the probability of each emotion based on the topic and user rating.

Step 3: Sentiment and emotion lexicon generation

The assumption that words in a document are the first indicator of the evoked emotion is assumed to be valid. However, the same word in different contexts may reflect different emotions, and words that bear emotional ambiguity are difficult to recognize out of context. Thus, other strategies are necessary to associate a sentiment or emotion with a given word. The POS of each word is used to alleviate the problem of emotional ambiguity of words and the context dependence of sentiment orientations. The POS of a word is a linguistic category defined by its syntactic or morphological behaviour. Categories include: noun, verb, adjective, adverb, pronoun, preposition, conjunction and interjection.

For example, the word ''bear'' has completely different orientations, one positive and one negative, in the following two sentences:

1. Teddy bear: a helping hand for disease sufferers.
2. They have to bear living with a disease.

The word ''bear'' is a noun in the first sentence and a verb in the second. A word feature f_j is defined as the association of the word W_j and its POS, e.g., (Kill/Verb). After defining the word feature f_j, its emotion probability is computed with equation (1):

$$EmoPro(e_i \mid f_j, t_k) = Val(f_j) \times \frac{\sum_{d \in C_{tk} \subset ND} p(f_j, t_k, d) \times oc(e_i, t_k)}{\sum_{e_i \in E} \left(\sum_{d \in C_{tk} \subset ND} p(f_j, t_k, d) \times oc(e_i, t_k) \right)} \quad (1)$$

where:

1. $Val(f_j)$ denotes the value (1 or 0) of word feature f_j in the intermediate lexicon.
2. $p(f_j, t_k, d)$ denotes the probability of feature f_j conditioned on document of corpus C_{tk} (subset of documents with topic t_k). $p(f_j, t_k, d)$ is the number of occurrences of the feature f_j in d divided by the total number of occurrences of all features in d.
3. $oc(e_i, t_k)$ denotes the co-occurrence number of documents d of C_{tk} and emotion e_i.

This strategy is used to eliminate emotions that are not associated with the same word in the NRC emotion lexicon. The sentiment probability of the word feature f_j is given by equation (2):

$$SemPro(s_i \mid f_j, t_k) = SSco(f_j) \times \frac{\sum_{d \in C_{tk} \subseteq ND} p(f_j, t_k, d) \times oc(s_i, t_k)}{\sum_{s_i \in S}\left(\sum_{d \in C_{tk} \subseteq ND} p(f_j, t_k, d) \times oc(s_i, t_k)\right)} \quad (2)$$

where:

1. $SSco(f_j)$ denotes the score of feature f_j in the intermediate lexicon.
2. $oc(s_i, t_k)$ denotes the co-occurrence number of documents d of C_{tk} and sentiment s_i.

Here, s_i may have two values, a positive sentiment S_P and negative sentiment S_N. Finally, to derive BMEmoWordMod, first the topic is added, then the emotion value is replaced by the computed emotion probability and the sentiment score with the computed sentiment probability.

2) Sentiment and emotion discovery - process phase

This phase identifies the sentiments and emotions that are likely associated with a given new document by using the sentiment and emotion semantic lexicon BMEmoWordMod generated in the previous section. After preprocessing, the term vector of the new document is defined using TF-IDF.

Let ND be the new document and $W_{ND} = \{W_1, \ldots, W_z\}$ the set of distinct terms occurring in the corpus of documents. To obtain the z-dimensional term vector that represents each document in the corpus, the tf-idf of each term of W_z is computed. The result of this computation establishes the term vector $\overrightarrow{t_{ND}} = (\text{tfidf}(W_1, ND), \ldots, \text{tfidf}(W_z, ND))$.

Using vector $\overrightarrow{t_{ND}}$, $T_{ND}=\{t_p, \ldots, t_q\}$ obtained using topic detection algorithm (assumed to be known) and BMEmoWordMod, the sentiment and emotion vector of new document $\overrightarrow{E_{f_j,ND}} = (E(f_j, ND, e_1), \ldots, E(f_j, ND, e_E), E(f_j, ND, s_P), E(f_j, ND, s_N))$ is given by equation (3):

$$E(f_j, ND, e_i) = \frac{\text{tfidf}(W_j, ND)}{\sum_{l=1}^{z} \text{tfidf}(W_l, ND)} \times \sum_{t_k \in T_{ND}} \text{BMEmoWord}(f_j, e_i, t_k) \quad (3)$$

where $\text{BMEmoWord}(f_j, e_i, t_k)$ denotes the emotion probability of emotion e_i for the feature word f_j giving the topic t_k. $\text{BMEmoWord}(f_j, e_i, t_k)$ is selected in BMEmoWordMod.

The weight of emotion e_i for document ND is computed with equation (4):

$$W_E(ND, e_i) = \sum_{W_j \in W_{ND}} E(f_j, ND, e_i) \quad (4)$$

Equation (4) yields the emotional vector of new document ND $\overrightarrow{V_{ND}} = (W_E(ND, e_1), \ldots, W_E(ND, e_i), \ldots, W_E(ND, e_E), W_E(ND, s_P), W_E(ND, s_N))$

Next, the new document ND emotion and sentiment is inferred using a fuzzy logic approach and the emotional vector $\overrightarrow{V_{ND}}$. The weight of emotion is transformed into five linguistic variables: very low, low, medium, high, and very high. Then, using these variables as input to the fuzzy inference system one obtains the final emotion for the new document. The fuzzy logic rules are predefined by experts.

3) Sentiment and emotion refining - process phase

The refining process validates discovered sentiment and emotion after the document analysis. Similarity is computed between new documents and documents in the corpus rated over E emotions. First, the term vectors of each document are defined using the tf-itf of each term, tf-itf is then computed using equation (5); to identify the most important terms of a given document D_j, the tf-idf of each term W_i in the corpus C_{ti} is computed using equation (5) as follows:

$$f(W_i, D_j, C_{ti}) = TF - IDF(W_i, D_j, C_{ti}) = TF(W_i, D_j) * \log\left(\frac{|C_{ti}| = M_i}{IDF(W_i, C_{ti})}\right) \quad (5)$$

Note that the terms extracted from the corpus of documents rated over E emotions are those employed by users. Next, to measure the similarity between two

documents, the cosine similarity of their representative vectors is computed using equation (6); given two documents $\overrightarrow{t_{d1}}$ and $\overrightarrow{t_{d2}}$, their cosine similarity is computed as:

$$SimCos\left(\overrightarrow{t_{d1}}, \overrightarrow{t_{d2}}\right) = \frac{\overrightarrow{t_{d1}} \cdot \overrightarrow{t_{d2}}}{\left|\overrightarrow{t_{d1}}\right| \times \left|\overrightarrow{t_{d2}}\right|} \quad (6)$$

Two documents d1 and d2 are similar when the similarity $SimCos\left(\overrightarrow{t_{d1}}, \overrightarrow{t_{d2}}\right)$ of these two documents is less than the similarity threshold β. Note that it is already assumed that when the similarity $SimCos\left(\overrightarrow{t_{d1}}, \overrightarrow{t_{d2}}\right)$ of two documents d1 and d2 is less than the similarity threshold β, the documents are not similar.

2. Evaluation using simulations

This section presents an evaluation of SSEA performance using simulations. To perform these simulations, an experimental environment called Libër was used. Libër was developed to provide a simulator to prototype the new algorithm SSEA.

G. Dataset and parameters

To evaluate SSEA, real datasets from different projects that have digital and physical library catalogues were used. These datasets, consisting of 25,000 documents with a vocabulary of 375,000 words, were selected using average TF-IDF for the analysis. The documents covered 20 topics and 8 emotions. The number of documents per topic or emotion was approximately equal. The average number of topics per document was 7 while the average rating emotion number per document was 4. 15,000 documents of the dataset were used for the training phase and the remaining 100 used for the test. Note that the 10,000 documents used for the tests were those that had more annotated topics or a higher rating over emotions.

To measure the performance of topic detection (sentiment and emotion discovery, respectively) approaches, comparison of detected topics (the discovered sentiment and emotion, respectively) with annotation topics of librarian experts (user ratings) were carried out. Table II presents the values of the parameters used in the simulations. The server characteristics for the simulations were: Dell Inc. PowerEdge R630 with 96 Ghz (4 x Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz, 10 core and 20 threads per CPU) and 256 GB memory running VMWare ESXi 6.0.

Table II: Simulation Parameters

Parameter	Value
ε	3
NumKeyTerm	8
ω	0.5
β	0.7
λ	0.6
α	100
co-occurrence threshold	0.75
semantic threshold	1
term cluster matching threshold	0.45

H. Performance criteria

SSEA performance was measured in terms of running time [16] and accuracy [42] [43]. Note that in the library domain, the most important criteria was precision while resource consumption was important for the software providers.

The running time, denoted by Rt, was computed as follows:

$$Rt = Et - Bt$$

where Et and denotes the time when processing is completed and Bt the time when it started.

To compute the accuracy, let E_{rating} and $E_{discovered}$ be the set of rating over emotion and the set of discovered emotion by SSEA for a given document d. The accuracy of sentiment and emotion discovery, denoted by A_d^e, was computed as follows:

$$A_d^e = \frac{2.\left|E_{\text{rating}} \cap E_{\text{discovered}}\right|}{\left|E_{\text{rating}}\right| + \left|E_{\text{discovered}}\right|}$$

Simulation results were averaged over multiple runs with different pseudorandom number generator seeds. The average accuracy, Ave_acc, of multiple runs was given by:

$$Ave\_acc = \frac{\sum_{x=1}^{I}\left(\frac{\sum_{d \in TD} A_d^e}{|TD|}\right)}{I}$$

where TD denotes the number of tests documents and I denotes the number of test iterations.

The average running time, *Ave_run_time*, was given by:

$$Ave\_run\_time = \frac{\sum_{x=1}^{I} Rt}{I}$$

I. Sentiment and emotion analysis performance evaluation

SSEA performance was also evaluated in terms of accuracy and running time. Simulations used the dataset and parameters mentioned previously. The performance of SSEA was compared to the approaches described in [34] and [37], referred to as ETM-LDA and AP, respectively. ETM-LDA and AP were selected because they were document-based rather than phrase-based.

1) Comparison of approaches with SSEA

Table III shows the characteristics of the approaches used for comparison with SSEA.

Table III: Sentiment and emotion approaches for comparison

Approach	Granularity	Approach	Training	Refining	Thesaurus	Topic	Emotions
AP [37]	D	L	Y	N	5	N	8
ETM-LDA [34]	D	K	Y	N	6	Y	8
SSEA	C	KR	Y	Y	1,2, 3,4	Y	8

1-WordNet; 2-WordNet-Affect; 3-SentiWordNet; 4-NRC Emotion Lexicon; 5- Stanford CoreNLP; 6-Gibbs sampling; D: Document; C: Configurable as desired; L: Learning based; K: Keyword based; KR: Keyword and Rule based; Y: Yes; N: No

SSEA was the only entirely semantic approach taking into account the rules for inferring emotion. In addition, SSEA used a semantic lexicon. Several approaches used semantic lexicon, but these were limited to phrases rather than documents. The best performance approaches used were AP and ETM_LDA.

2) Results analysis

Figure 3 presents the average running time when varying the number of detected emotions. Training times were excluded because this phase was performed only once. The SSEA training phase took more time than the other approaches due to lexicon aggregation and enrichment by users. The average running time increased with the number of test documents. This is normal, as the larger the number of test documents the longer the average running time to perform the sentiment and emotion discovery.

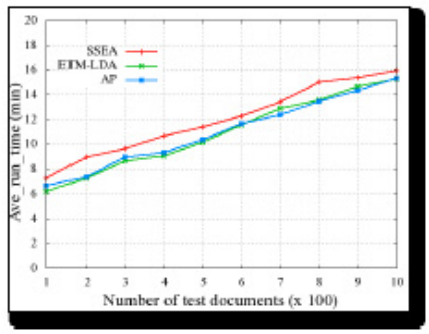


Figure 3: Emotion discovery - Average running time versus number of documents for test phase

Figure 3 shows that ETM-LDA and AP outperformed SSEA on the running time criteria. ETM-LDA required an average of 1.53 sec per document whereas SSEA required an average of 1.74 sec per document. The average relative improvement of ETM-LDA compared with SSEA was approximately 0.21 sec per document. The poorer performance of SSEA resulted from refining sentiment and emotion to increase accuracy.

Figure 4 presents the average accuracy when varying the number of discovered emotions.

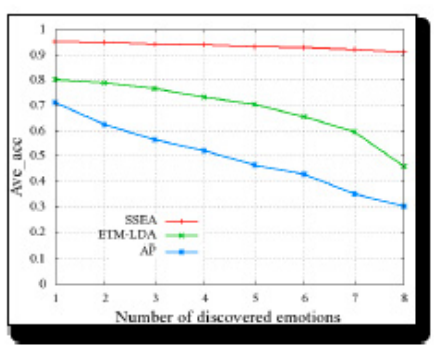


Figure 4: Average detection accuracy for the number of discovered emotions

Positive and negative sentiments were not considered in the accuracy measurement. Figure 4 also shows that the average accuracy decreased with the number of discovered emotions. However, SSEA outperformed the other two approaches used for comparisons. SSEA demonstrated an average accuracy of 93.30% per emotion while ETM-LDA, the best of the other two approaches used for comparison, produced 68.65% accuracy per emotion. The average relative improvement in accuracy of SSEA compared to ETM-LDA was 24.65% per emotion.

In conclusion, the 0.21 sec running time per document increase was, again, a small price to pay for the larger average accuracy of emotion discovery (24.65%).

IV. CONCLUSION

Following is our conclusions on related work in sentiment and emotion analysis:

1. Traditional sentiment analysis methods mainly use terms and their frequency, part of speech, rule of opinions and sentiment shifters. Semantic information is ignored in term selection, and it is difficult to find complete rules.
2. Most of the recent contributions are limited to sentiment analysis elaborated in terms of positive or negative opinion and do not include analysis of emotion.
3. Existing approaches do not take into account the human contribution to improve accuracy.
4. Existing approaches do not combine sentiment and emotion analysis.
5. Lexicon and ontology based approaches provide good accuracy for text-based sentiment and emotion analysis when applying SVM techniques. In our work, it is more important to identify the sentiment and emotion of a book taking into account all the books of the collection. For example, assume that book A has 90% fear and 80% sadness while the emotion which has the best weight of book B is 40% fear; can it be said that fear is the emotion of book B as in book A?
6. Existing approaches do not take into account document collections. In terms of granularity, most of the existing approaches are sentence-based.
7. These approaches do not take into account the context around the sentence and in this way, it is possible to lose the real emotion.

As a general conclusion to the literature review on topic detection, sentiment and emotion analysis, 95% of the work focused on features of the documents (e.g., sentence length, capitalized words, document title, term frequency, and sentences position) to perform text mining and generally make use of existing algorithms or approaches (e.g., LDA, tf-idf, VSM, SVD, LSA, TextRank, PageRank, LexRank, FCA, LTM, SVM, NB and ANN) based on their associated features to documents.

Table I compares the most known text mining algorithms (e.g., AlchemyAPI, DBpedia, Wikimeta, open calais, Bitext, AIDA, TextRazor) with our proposed algorithm in SMESE by keyword extraction, classification, sentiment analysis, emotion analysis and concept extraction.

V. SUMMARY AND FUTURE WORK

In this paper, the goal was to increase the findability (search, discover) of entities based on user interest using external and internal semantic metadata enrichment algorithms. As computers struggle to understand the meaning of natural language, enriching entities semantically with meaningful metadata can improve search engine capability. Words themselves have a wide variety of definitions and interpretations and are often utilized inconsistently. While sentiment and emotion may have no relationship to individual words, thesauri express associative relationships between words, ontologies, entities and a multitude of relationships represented as triplets.

This paper presented an enhanced implementation of SMESE [2] and SSEA algorithm based on text analysis approaches. It includes distinct task that:

1. Discover enriched sentiment and emotion metadata hidden within the text or linked to multimedia structure using the proposed SSEA (Semantic Sentiment and Emotion Analysis) algorithm.
2. Implement rule-based semantic metadata internal enrichment includes algorithm named SSEA.

Table I shows the comparison with most known text mining algorithms (e.g., AlchemyAPI, DBpedia, Wikimeta, Open Calais, Bitext, AIDA, TextRazor) and a new algorithm SSEA with many attributes including keyword extraction, classification, sentiment analysis, emotion analysis, and concept extraction. It was noted

that this algorithm supports more attributes than any other algorithms.

In future work, the focus will be to connect emotion and sentiment to the users evolving interests and will include:

1. Some enhancements to be able to enrich metadata semantically, including the evolution of the user interests over time.
2. Further evaluations of the SSEA model and algorithm with different prototype and datasets.

Exploring text summarization and automatic literature review as metadata enrichments.

VI. REFERENCES

[1]. O. Appel, F. Chiclana, J. Carter, and H. Fujita, "A hybrid approach to the sentiment analysis problem at the sentence level," Knowledge-Based Systems, vol. 108, pp. 110-124, 2016. doi:http://dx.doi.org/10.1016/j.knosys.2016.05.040

[2]. R. Brisebois, A. Abran, and A. Nadembega, "A Semantic Metadata Enrichment Software Ecosystem (SMESE) based on a Multi-platform Metadata Model for Digital Libraries," Accepted for publication in Journal of Software Engineering and Applications (JSEA), vol. 10, no. 04, 2017

[3]. G. Salton, and C. Buckley, "Term-weighting approaches in automatic text retrieval," Information Processing & Management, vol. 24, no. 5, pp. 513-523, 1988. doi:http://dx.doi.org/10.1016/0306-4573(88)90021-0

[4]. T. Niu, S. Zhu, L. Pang, and A. El Saddik, "Sentiment Analysis on Multi-View Social Data," in 22nd International Conference on MultiMedia Modeling (MMM), Miami, FL, USA, 2016, pp. 15-27. doi:http://dx.doi.org/10.1007/978-3-319-27674-8_2

[5]. K. Bougiatiotis, and T. Giannakopoulos, "Content Representation and Similarity of Movies based on Topic Extraction from Subtitles," in Proceedings of the 9th Hellenic Conference on Artificial Intelligence, Thessaloniki, Greece, 2016, pp. 1-7. doi:http://dx.doi.org/10.1145/2903220.2903235

[6]. G. A. Patel, and N. Madia, "A Survey: Ontology Based Information Retrieval For Sentiment Analysis," International Journal of Scientific Research in Science, Engineering and Technology, vol. 2, no. 2, pp. 460-465, 2016

[7]. J. A. Balazs, and J. D. Velásquez, "Opinion Mining and Information Fusion: A survey," Information Fusion, vol. 27, pp. 95-110, 2016. doi:http://dx.doi.org/10.1016/j.inffus.2015.06.002

[8]. K. Ravi, and V. Ravi, "A survey on opinion mining and sentiment analysis: Tasks, approaches and applications," Knowledge-Based Systems, vol. 89, pp. 14-46, 2015. doi:http://dx.doi.org/10.1016/j.knosys.2015.06.015

[9]. J. Serrano-Guerrero, J. A. Olivas, F. P. Romero, and E. Herrera-Viedma, "Sentiment analysis: A review and comparative analysis of web services," Information Sciences, vol. 311, pp. 18-38, 2015. doi:http://dx.doi.org/10.1016/j.ins.2015.03.040

[10]. M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," Comput. Linguist., vol. 37, no. 2, pp. 267-307, 2011. doi:10.1162/COLI_a_00049

[11]. D. Vilares, M. A. Alonso, and C. GÓMez-RodrÍGuez, "A syntactic approach for opinion mining on Spanish reviews," Natural Language Engineering, vol. 21, no. 1, pp. 139-163, 2015. doi:http://dx.doi.org/10.1017/S1351324913000181

[12]. S. Kiritchenko, X. Zhu, and S. M. Mohammad, "Sentiment analysis of short informal texts," Journal of Artificial Intelligence Research, vol. 50, no. 1, pp. 723-762, 2014. doi:http://dx.doi.org/10.1613/jair.4272

[13]. D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," Journal of Machine Learning Research, vol. 3, pp. 993–1022, 2003

[14]. S. T. Dumais, "Latent semantic analysis," Annual Review of Information Science and Technology, vol. 38, no. 1, pp. 188-230, 2004. doi:10.1002/aris.1440380105

[15]. J. Cigarrán, Á. Castellanos, and A. García-Serrano, "A step forward for Topic Detection in Twitter: An FCA-based approach," Expert Systems with Applications, vol. 57, pp. 21-36, 2016. doi:http://dx.doi.org/10.1016/j.eswa.2016.03.011

[16]. P. Chen, N. L. Zhang, T. Liu, L. K. M. Poon, and Z. Chen, "Latent Tree Models for Hierarchical Topic Detection," arXiv preprint arXiv:1605.06650 cs.CL], pp. 1-44, 2016

[17]. R. Moraes, J. F. Valiati, and W. P. Gavião Neto, "Document-level sentiment classification: An empirical comparison between SVM and ANN," Expert Systems with Applications, vol. 40, no. 2, pp. 621-633, 2013. doi:http://dx.doi.org/10.1016/j.eswa.2012.07.059

[18]. M. Ghiassi, J. Skinner, and D. Zimbra, "Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network," Expert Systems with Applications, vol. 40, no. 16, pp. 6266-6282, 2013. doi:http://dx.doi.org/10.1016/j.eswa.2013.05.057

[19]. A. Gangemi, "A Comparison of Knowledge Extraction Tools for the Semantic Web," in 10th European Semantic Web Conference (ESWC), Montpellier, France, 2013, pp. 351-366. doi:http://dx.doi.org/10.1007/978-3-642-38288-8_24

[20]. S. N. Shivhare, and S. Khethawat, "Emotion Detection from Text," in Second International Conference on Computer Science, Engineering and Applications (ICCSEA), Delhi, India, 2012, pp. 1-7

[21]. A. Moreo, M. Romero, J. L. Castro, and J. M. Zurita, "Lexicon-based Comments-oriented News Sentiment Analyzer system," Expert Systems with Applications, vol. 39, no. 10, pp. 9166-9180, 2012. doi:http://dx.doi.org/10.1016/j.eswa.2012.02.057

[22]. C. Bosco, V. Patti, and A. Bolioli, "Developing corpora for sentiment analysis: The case of irony and senti-tut," IEEE Intelligent Systems, vol. 28, no. 2, pp. 55-63, 2013

[23]. H. Cho, S. Kim, J. Lee, and J.-S. Lee, "Data-driven integration of multiple sentiment dictionaries for lexicon-based sentiment classification of product reviews," Knowledge-Based Systems, vol. 71, pp. 61-71, 2014. doi:http://dx.doi.org/10.1016/j.knosys.2014.06.001

[24]. C. Lin, Y. He, R. Everson, and S. Ruger, "Weakly Supervised Joint Sentiment-Topic Detection from Text," IEEE Transactions on Knowledge and Data Engineering, vol. 24, no. 6, pp. 1134-1145, 2012. doi:http://dx.doi.org/10.1109/TKDE.2011.48

[25]. E. Kontopoulos, C. Berberidis, T. Dergiades, and N. Bassiliades, "Ontology-based sentiment analysis of twitter posts," Expert Systems with Applications, vol. 40, no. 10, pp. 4065-4074, 2013. doi:http://dx.doi.org/10.1016/j.eswa.2013.01.001

[26]. B. Desmet, and V. Hoste, "Emotion detection in suicide notes," Expert Systems with Applications, vol. 40, no. 16, pp. 6351-6358, 2013. doi:http://dx.doi.org/10.1016/j.eswa.2013.05.050

[27]. M. Abdul-Mageed, M. Diab, and S. Kübler, "SAMAR: Subjectivity and sentiment analysis for Arabic social media," Computer Speech & Language, vol. 28, no. 1, pp. 20-37, 2014. doi:http://dx.doi.org/10.1016/j.csl.2013.03.001

[28]. L. K.-W. Tan, J.-C. Na, Y.-L. Theng, and K. Chang, "Phrase-Level Sentiment Polarity Classification Using Rule-Based Typed Dependencies and Additional Complex Phrases Consideration," Journal of Computer Science and Technology, vol. 27, no. 3, pp. 650-666, 2012. doi:http://dx.doi.org/10.1007/s11390-012-1251-y

[29]. L. Chen, L. Qi, and F. Wang, "Comparison of feature-level learning methods for mining online consumer reviews," Expert Systems with Applications, vol. 39, no. 10, pp. 9588-9601, 2012. doi:http://dx.doi.org/10.1016/j.eswa.2012.02.158

[30]. C. Quan, and F. Ren, "Unsupervised product feature extraction for feature-oriented opinion determination," Information Sciences, vol. 272, pp. 16-28, 2014. doi:http://dx.doi.org/10.1016/j.ins.2014.02.063

[31]. S. Poria, E. Cambria, A. Hussain, and G.-B. Huang, "Towards an intelligent framework for multimodal affective data analysis," Neural Networks, vol. 63, pp. 104-116, 2015. doi:http://dx.doi.org/10.1016/j.neunet.2014.10.005

[32]. M. D. Munezero, C. S. Montero, E. Sutinen, and J. Pajunen, "Are They Different? Affect, Feeling, Emotion, Sentiment, and Opinion Detection in Text," IEEE Transactions on Affective Computing, vol. 5, no. 2, pp. 101-111, 2014. doi:http://dx.doi.org/10.1109/TAFFC.2014.2317187

[33]. W. Li, and H. Xu, "Text-based emotion classification using emotion cause extraction," Expert Systems with Applications, vol. 41, no. 4, Part 2, pp. 1742-1749, 2014. doi:http://dx.doi.org/10.1016/j.eswa.2013.08.073

[34]. S. Bao, S. Xu, L. Zhang, R. Yan, Z. Su, D. Han, and Y. Yu, "Mining Social Emotions from Affective Text," IEEE Transactions on

Knowledge and Data Engineering, vol. 24, no. 9, pp. 1658-1670, 2012. doi:http://dx.doi.org/10.1109/TKDE.2011.188

[35]. S. V. Kedar, D. S. Bormane, A. Dhadwal, S. Alone, and R. Agarwal, "Automatic Emotion Recognition through Handwriting Analysis: A Review," in 2015 International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2015, pp. 811-816. doi:http://dx.doi.org/10.1109/ICCUBEA.2015.162

[36]. J. Lei, Y. Rao, Q. Li, X. Quan, and L. Wenyin, "Towards building a social emotion detection system for online news," Future Generation Computer Systems, vol. 37, pp. 438-448, 2014. doi:http://dx.doi.org/10.1016/j.future.2013.09.024

[37]. V. Anusha, and B. Sandhya, "A Learning Based Emotion Classifier with Semantic Text Processing," Advances in Intelligent Informatics, M. E.-S. El-Alfy, M. S. Thampi, H. Takagi, S. Piramuthu and T. Hanne, eds., pp. 371-382, Cham, Switzerland: Springer International Publishing, 2015. doi:http://dx.doi.org/10.1007/978-3-319-11218-3_34

[38]. E. Cambria, P. Gastaldo, F. Bisio, and R. Zunino, "An ELM-based model for affective analogical reasoning," Neurocomputing, vol. 149, Part A, pp. 443-455, 2015. doi:http://dx.doi.org/10.1016/j.neucom.2014.01.064

[39]. M. F. Porter, "An algorithm for suffix stripping," Program, vol. 14, no. 3, pp. 130-137, 1980. doi:doi:10.1108/eb046814

[40]. de Marneffe M-C, MacCartney B, and Manning CD, "Generating typed dependency parsers from phrase structure parses " in fifth international conference on language resources and evaluation, GENOA , ITALY 2006, pp. 449–54

[41]. Y. Tsuruoka, and J. i. Tsujii, "Bidirectional inference with the easiest-first strategy for tagging sequence data," in Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, British Columbia, Canada, 2005, pp. 467-474. doi:10.3115/1220575.1220634

[42]. C. Zhang, H. Wang, L. Cao, W. Wang, and F. Xu, "A hybrid term–term relations analysis approach for topic detection," Knowledge-Based Systems, vol. 93, pp. 109-120, 2016. doi:http://dx.doi.org/10.1016/j.knosys.2015.11.006

[43]. H. Sayyadi, and L. Raschid, "A Graph Analytical Approach for Topic Detection," ACM Transactions on Internet Technology, vol. 13, no. 2, pp. 1-23, 2013. doi:http://dx.doi.org/10.1145/2542214.2542215

Paper 4:

A Semantic Metadata Enrichment Software Ecosystem based on Topic Metadata Enrichments

Ronald Brisebois, Alain Abran, Apollinaire Nadembega, Philippe N'techobo

http://aircconline.com/ijdkp/V7N3/7317ijdkp01.pdf

International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol.7, No.3, May 2017

A SEMANTIC METADATA ENRICHMENT SOFTWARE ECOSYSTEM BASED ON TOPIC METADATA ENRICHMENTS

Ronald Brisebois[1], Alain Abran[1], Apollinaire Nadembega[2] and Philippe N'techobo[3]

[1]École de technologie supérieure, University of Quebec, Montreal, Canada
[2]Network Research Lab., University of Montreal, Montreal, Canada
[3]École Polytechnique de Montréal, Montreal, Canada

ABSTRACT

As existing computer search engines struggle to understand the meaning of natural language, semantically enriched metadata may improve interest-based search engine capabilities and user satisfaction.

This paper presents an enhanced version of the ecosystem focusing on semantic topic metadata detection and enrichments. It is based on a previous paper, a semantic metadata enrichment software ecosystem (SMESE). Through text analysis approaches for topic detection and metadata enrichments this paper propose an algorithm to enhance search engines capabilities and consequently help users finding content according to their interests. It presents the design, implementation and evaluation of SATD (Scalable Annotation-based Topic Detection) model and algorithm using metadata from the web, linked open data, concordance rules, and bibliographic record authorities. It includes a prototype of a semantic engine using keyword extraction, classification and concept extraction that allows generating semantic topics by text, and multimedia document analysis using the proposed SATD model and algorithm.

The performance of the proposed ecosystem is evaluated using a number of prototype simulations by comparing them to existing enriched metadata techniques (e.g., AlchemyAPI, DBpedia, Wikimeta, Bitext, AIDA, TextRazor). It was noted that SATD algorithm supports more attributes than other algorithms. The results show that the enhanced platform and its algorithm enable greater understanding of documents related to user interests.

KEYWORDS

Natural Language Processing, Semantic Topic Detection, Semantic Metadata Enrichment, Text and Data Mining

1. INTRODUCTION

The goal of this paper is to increase the findability of document or content matching user interest using an internal semantic metadata enrichment algorithm. Words themselves are often used inconsistently, having a wide variety of definitions and interpretations. Finding bibliographic references or semantic relationships in texts makes it possible to localize specific text segments using ontologies to enrich a set of semantic metadata related to topics. This paper presents an enhanced implementation of SMESE [1] focusing on semantic topic metadata detection and enrichment.

Semantic topic detection (STD), a fundamental aspect of SIR, helps users to efficiently detect meaningful topics. Initial methods for STD relied on clustering documents based on a core group of keywords representing a specific topic, where, based on a ratio such as tf-idf, documents that contain these keywords are similar to each other [2,3]. Next, variations of tf-idf were used to compute keyword-based feature values, and cosine similarity was used as a similarity (or distance) measure to

DOI: 10.5121/ijdkp.2017.7301

cluster documents. The following generation of STD approaches, including those based on latent Dirichlet allocation (LDA), shifted analysis from directly clustering documents to clustering keywords. Some examples of these advances in STD are presented in [4]. Bijalwan et al. [5], for example, experimented with machine learning approaches for text and document mining and concluded that k-nearest neighbors (KNN), for their data sets, showed the maximum accuracy as compared to naive Bayes and term-graph. The drawback for KNN is that time load is high but it demonstrates better accuracy than others.

A number of approaches are used to perform text mining, including: latent Dirichlet allocation (LDA) [4], tf-idf [2,3], latent semantic analysis (LSA) [6], formal concept analysis (FCA) [7], latent tree model (LTM) [8], naïve Bayes (NB) [9], and artificial neural network (ANN) [10]. This paper consists of a model and an algorithm SATD (Scalable Annotation-based Topic Detection) for topic metadata semantic enrichments. SATD allows the generation of semantic topics using text, relationships and documents analysis. Using simulation, the performance of SATD was evaluated in terms of accuracy of topic detection. For comparison, existing approaches that performs semantic metadata enrichment in terms of topic detection and enrichment were evaluated. Simulation results showed that SATD outperforms these existing approaches.

The remainder of the paper is organized as follows. Section 2 presents the related work. Section 3 describes SATD model and algorithm while Section 4 presents the evaluation through different prototypes. Section 5 concludes the paper and presents some future work.

2. RELATED WORK

Generally, a topic is represented as a set of descriptive and collocated keywords/terms. Initially, document clustering techniques were adopted to cluster content-similar documents and extract keywords from clustered document sets as the representation of topics. The predominant method for topic detection is the latent Dirichlet allocation (LDA) [4], which assumes a generating process for the documents. LDA has been proven a powerful algorithm because of its ability to mine semantic information from text data. Terms having semantic relations with each other are collected as a topic. LDA is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities.

The literature presents two groups of text-based topic detection approaches based on the size of the text: short text [11,7,12,13] such as tweets or Facebook posts, and long text [14,4,15-17,8] such as a document or a book. For example, Dang et al. [11] proposed an early detection method for emerging topics based on dynamic Bayesian networks in micro-blogging networks. They analyzed the topic diffusion process and identified two main characteristics of emerging topics, namely attractiveness and key-node. Next, based on this identification, they selected features from the topology properties of topic diffusion, and built a DBN-based model using the conditional dependencies between features to identify the emerging keywords. But to do so, they had to create a term list of emerging keyword candidates by term frequency in a given time interval. Cigarran et al. [7] proposed an approach based on formal concept analysis (FCA). Formal concepts are conceptual representations based on the relationships between tweet terms and the tweets that have given rise to them. Cotelo et al. [12], when addressing the tweet categorization task, explored the idea of integrating two fundamental aspects of a tweet: the textual content itself, and its underlying structural information. This work focuses on long text topic detection.

Recently, considerable research has gone into developing topic detection approaches using a number of information extraction techniques (IET), such as lexicon, sliding window, boundary techniques, etc. Many of these techniques [14,15,17,8] rely heavily on simple keyword extraction from text. For example, Sayyadi and Raschid [14] proposed an approach for topic detection, based on keyword-based methods, called KeyGraph, that was inspired by the keyword co-occurrence graph and efficient graph analysis methods. The main steps in the KeyGraph approach are as follows:

1. The first step is construction of a keyword co-occurrence graph, called a KeyGraph, which has one node for each keyword in the corpus and where edges represent the co-occurrence of the corresponding keywords weighted by the count of the co-occurrences.
2. Secondly, making use of an off-the-shelf community detection algorithm, community detection is taken into account where each community forms a cluster of keywords that represent a topic. The weight of each keyword in the topic feature vector is computed using the tf-idf formula. The TF value is computed as the average co-occurrence of each keyword from the community with respect to the other keywords in that community.
3. Then, to assign a topic to a document, the likelihood of each topic t with the vector of keyword f_t is computed using the cosine similarity of the document.
4. Finally, for each pair of topics, where multiple documents are assigned to both topics, it is assumed that these are subtopics of the same parent topic and are therefore merged.

In other words, KeyGraph is based on the similarity of keyword extraction from text. We note two limitations to the approach, which requires improvement in two respects. Firstly, they failed to leverage the semantic information derived from topic model. Secondly, they measured co-occurrence relations from an isolated term-term perspective; that is, the measurement was limited to the term itself and the information context was overlooked, which can make it impossible to measure latent co-occurrence relations. Salatino and Motta [17] suggested that it is possible to forecast the emergence of novel research topics even at an early stage and demonstrated that such an emergence can be anticipated by analyzing the dynamics of pre-existing topics. They presented a method that integrates statistics and semantics for assessing the dynamics of a topic graph: (1) first, they select and extract portions of the collaboration networks related to topics in the two groups a few years prior to the year of analysis. Based on these topics, they build a topics graph where nodes are the keywords while edges are the links representing co-occurrences between keywords and (2) next, they transform the graphs into sets of 3-cliques. For each node of a 3-clique, they compute the weight associated with each link between pairs of topics by using the harmonic mean of the conditional probabilities. While this is a satisfactory approach to find latent co-occurrence relations, the approach assumes that keywords are topics. Chen et al. [8] proposed a novel method for hierarchical topic detection where topics are obtained by clustering documents in multiple ways. They used a class of graphical models called hierarchical latent tree models (HLTMs). Latent tree models (LTMs) are tree-structured probabilistic graphical models where the variables at leaf nodes are observed and the variables at internal nodes are latent. It is a Markov random field over an undirected tree carried out as follows: (1) first, the word variables are partitioned into clusters such that the words in each cluster tend to co-occur and the co-occurrences can be properly modeled using a single latent variable. The authors achieved this partition using the BUILDISLANDS subroutine, which is based on a statistical test called the uni-dimensionality test (UD-test) and (2) after the islands are created, they are linked up so as to obtain a model over all the word variables. This is carried out by the BRIDGEISLANDS subroutine, which estimates the mutual information between each pair of latent variables in the islands. This allows construction of a complete undirected graph with the mutual information values as edge weights, and finally the maximum spanning tree of the graph is determined [8]. Hurtado et al. [18] proposed an approach that uses sentence-level association rule mining to discover topics from documents. Their method considers each sentence as a transaction and keywords within the sentence as items in the transaction. By exploring keywords (frequently co-occurring) as patterns, their method preserves contextual information in the topic mining process. For example, whenever the terms: "machine", "support" and "vector" are discovered as strongly correlated keywords, either as "support vector machine" or "support vector", they assumed that these patterns were related to one topic, i.e., "SVM". In order to discover a set of strongly correlated topics, they used the CPM-based community detection algorithm to find groups of topics with strong correlations. As in [8], their contribution was limited to simulating existing algorithms. Zhang et al. [15] proposed LDA-IG, an extension of KeyGraph [14]. It is a hybrid relations analysis approach integrating semantic relations and co-occurrence relations for topic detection. Specifically, their approach fuses multiple types of relations into a uniform term graph by incorporating idea discovery theory with a topic modeling method.

1. Firstly, they defined an idea discovery algorithm called IdeaGraph that was adopted to mine latent co-occurrence relations in order to convert the corpus into a term graph.
2. Next, they proposed a semantic relation extraction approach based on LDA that enriches the graph with semantic information.
3. Lastly, they make use of a graph analytical method to exploit the graph for detecting topics. Their approach has four steps: (a) Pre-processing to filter noise and adjust the data format suitable for the subsequent components, (b) Term graph generation to convert the basket dataset into a term graph by extracting co-occurrence relations between terms using the Idea Discovery algorithm, (c) Term graph refining with semantic information using LDA to build semantic topics and tp-izp, inspired by tf-idf, to measure the semantic value of any term in each topic, and (d) Topic extraction from the refined term graph by assuming that a topic is a filled polygon and measuring the likelihood of a document d being assigned to a topic using tf-idf. However, their approach does not include machine learning.

From our review of related work, we conclude that the main drawbacks of existing approaches to topic detection are as follows:

1. They are based on simple keyword extraction from text and lack semantic information that is important for understanding the document. To tackle this limitation, our work uses semantic annotations to improve document comprehension time.
2. Co-occurrence relations across the document are commonly neglected, which leads to incomplete detection of information. Current topic modeling methods do not explicitly consider word co-occurrences because of a computational challenge. The graph analytical approach to this extension was only an approximation that merely took into account co-occurrence information alone while ignoring semantic information. How to combine semantic relations and co-occurrence relations to complement each other remains a challenge.
3. Existing approaches focus on detecting prominent or distinct topics based on explicit semantic relations or frequent co-occurrence relations; as a result, they ignore latent co-occurrence relations. In other words, latent co-occurrence relations between two terms cannot be measured from an isolated term-term perspective. The context of the term needs to be taken into account.
4. More importantly, even though existing approaches take into account semantic relations, they do not include machine learning to find new topics automatically.

The main conclusion is that most of the existing related research is limited to simulations using existing algorithms. None contribute improvements to detect topics more accurately.

Table 1 compares the most known text mining algorithms (e.g., AlchemyAPI, DBpedia, Wikimeta, Bitext, AIDA, TextRazor) with our proposed algorithm in SMESE V3 by keyword extraction, classification and concept extraction.

Table 1. Summary of attribute comparison of existing and SATD algorithms.

Existing algorithms	Keyword extraction	Classification	Concept extraction
AlchemyAPI (http://www.alchemyapi.com/)	x	x	x
DBpedia Spotlight (https://github.com/dbpedia-spotlight)			x
Wikimeta (https://www.w3.org/2001/sw/wiki/Wikimeta)			x
Yahoo! Content Analysis API (out of date) (https://developer.yahoo.com/contentanalysis/)		x	x
Tone Analyzer (https://tone-analyzer-demo.mybluemix.net/)			
Zemanta (http://www.zemanta.com/)			x
Receptiviti (http://www.receptiviti.ai/)			
Apache Stanbol (https://stanbol.apache.org/)			x
Bitext (https://www.bitext.com/)			x

Mood patrol (https://market.mashape.com/ soulhackerslabs/ mood patrol-emotion-detection-from-text)			
Aylien (http://aylien.com/)	x	x	
AIDA (http://senseable.mit.edu/aida/)			x
Wikifier (http://wikifier.org/)			x
TextRazor (https://www.textrazor.com/)			x
Synesketch (http://krcadinac.com/synesketch/)			
Toneapi (http://toneapi.com/)			
SATD algorithm	x	x	x

3. RULE-BASED SEMANTIC METADATA INTERNAL ENRICHMENT ENGINE

This section presents an overview and details of the proposed rule-based semantic metadata internal enrichment engine, including the model and algorithm (SATD) used to process semantic metadata internal enrichment for topic.

The goal of this paper is to extend the SMESE platform [1] through text analysis approaches for topic detection and metadata enrichments. To perform this task, the following tools are needed: (1) topics are a controlled set of terms designed to describe the subject of a document. While topics do not necessarily include relationships between terms, we include relationships as triplets (Entity – Relationship – Entity); for example, Entity "*Ronald*" - relationship:" *likes* " - Entity "*Le petit prince*", and (2) an ontology to provide a representation of knowledge with rich semantic relationships between topics. By breaking content into pieces of data, and curating semantic relationships to external contents, metadata enrichments are created dynamically.

3.1. Rule-based semantic metadata internal enrichment engine overview

The rule-based semantic metadata internal enrichment engine has been designed to find short descriptions, in terms of topics of the members of a collection to enable efficient processing of large collections while preserving the semantic and statistical relationships. Figure 1 shows an overview of the architecture that consists of: (1) User interest-based gateway, (2) Metadata initiatives & concordance rules, (3) Harvesting web metadata & data, (4) User profiling engine and (5) Rule-based semantic metadata internal enrichment engine. The user interest-based gateway is designed to push notifications to users based on the topics found using the user-profiling engine. The rule-based semantic metadata internal enrichment engine performs automated metadata internal enrichment based on the set of metadata initiatives & concordance rules, the engine for harvesting web metadata, the user profile and a thesaurus.

The following sub-sections present the terminology and assumptions, and details of the SATD algorithm.

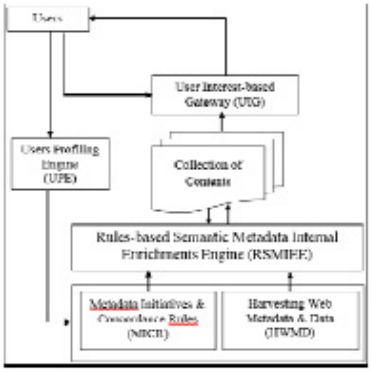


Figure 1. Rule-based semantic metadata internal enrichment engine architecture

3.2. Terminology and assumptions

In this section the following terms are defined:

1. A word or term is the basic unit of discrete data, defined to be an item from a vocabulary indexed by {1, …,V}. Terms are presented using unit-basis vectors that have a single component equal to one and all other components equal to zero. Thus, using superscripts to denote components, the ith term in the vocabulary is represented by an I-vector w such that $w^i = 1$ and $w^j = 0$ for $i \neq j$. For example, let V= {book, image, video, cat, dog} be the vocabulary. The video term is represented by the vector (0, 0, 1, 0, 0).
2. A line is a sequence of N terms denoted by l. These terms are extracted from a real sentence; a sentence is a group of words, usually containing a verb, that expresses a thought in the form of a statement, question, instruction, or exclamation and when written begins with a capital letter.
3. A document is a sequence of N lines denoted by $D = (w_1, w_2; \ldots, w_N)$, where w_i is the ith term in the sequence coming from the lines. D is represented by its lines as $D = (l_1, \ldots l_i, \ldots, l_K)$.
4. A corpus is a collection of M documents denoted by $C = \{D_1, D_2, \ldots, D_M\}$.
5. An emotion word is a word with strong emotional tendency. An emotion word is a probabilistic distribution of emotions and represents a semantically coherent emotion analysis. For example, the word "excitement", presenting a positive and pleased feeling, is assigned a high probability to emotion "joy".

To implement the SATD algorithm, an initial set of conditions must be established:

1. A list of topics $T = \{t_1, \ldots, t_i, \ldots, t_n\}$ is readily available.
2. Each existing document D_j is already annotated by topic. The annotated topics of document D_j are denoted as $T_{Dj} = \{t_p \ldots, t_i, \ldots, t_q\}$ where t_p, t_i, and $t_q \in T$.
3. The corpus of documents is already classified by topics. $C_{ti} = \{\ldots, D_j, \ldots\}$ denotes the corpus of documents that have been annotated with topic t_i. Note that the document D_j may be located in several corpuses.
4. A list of sentiments $S = \{s_1, \ldots, s_i, \ldots, s_S\}$ is readily available.
5. A thesaurus is available and has a tree hierarchical structure.

3.3. Document pre-processing

The objective of the pre-processing is to filter noise and adjust the data format to be suitable for the analysis phases. It consists of stemming, phase extraction, part-of-speech filtering and removal of stop-words. The corpus of documents crawled from specific databases or the internet consists of many documents. The documents are pre-processed into a basket dataset C, called document collection. C consists of lines representing the sentences of the documents. Each line consists of terms, i.e. words or phrases. 'Word' and 'term' are used interchangeably in the rest of this paper.

More specifically, to obtain D_j, the following preprocessing steps are performed: (1) Language detection, (2) Segmentation: a process of dividing a given document into sentences, (3) Stop word: a process to remove the stop words from the text. Stop words are frequently occurring words such as 'a' an', the' that provide less meaning and generate noise. Stop words are predefined and stored in an array, (4) Tokenization: separates the input text into separate tokens, (5) Punctuation marks: identifies and treats the spaces and word terminators as the word breaking characters, and (6) Word stemming: converts each word into its root form by removing its prefix and suffix for comparison with other words. More specifically, a standard preprocessing such as tokenization, lowercasing and stemming of all the terms using the Porter stemmer [19]. Therefore, we also parse the texts using the Stanford parser [20] that is a lexicalized probabilistic parser which provides various information such as the syntactic structure of text segments, dependencies and POS tags.

3.4. Scalable annotation-based topic detection: SATD

The aim of SATD is to build a classifier that can learn from already annotated documents and infer the topics. Traditional approaches are typically based on various topic models, such as latent Dirichlet allocation (LDA) where authors cluster terms into a topic by mining semantic relations between terms. Furthermore, the inability to discover latent co-occurrence relations via the context or other bridge

terms prevents important but rare topics from being detected. SATD combines semantic relations between terms and co-occurrence relations across the document making use of document annotation. In addition, SATD includes: (1) a probabilistic topic detection approach that is an extension of LDA, called BM semantic topic model (BM-SemTopic) and (2) a clustering approach that is an extension of KeyGraph, called BM semantic graph (BM-SemGraph).

SATD is a hybrid relation analysis and machine learning approach that integrates semantic relations, semantic annotations and co-occurrence relations for topic detection. More specifically, SATD fuses multiple relations into a term graph and detects topics from the graph using a graph analytical method. It can detect topics not only more effectively by combing mutually complementary relations, but also mine important rare topics by leveraging latent co-occurrence relations.

SATD is composed of five phases: (1) relevant and less similar documents selection process phase, (2) not annotated documents semantic term graph generation process phase, (3) topics detection process phase, (4) training process phase and (5) topics refining process phase. The following sub-sections present the details of the five phases of the SATD model.

3.4.1. Relevant and less similar documents selection - process phase

For a given topic, a filtering process is performed to avoid using a large corpus of documents that are similar or not relevant. For this reason, only relevant and less similar documents within a corpus are identified. Here, only documents that are already annotated by topic are considered.
An overview of the architecture of the relevant and less similar document selection phase is presented in Figure 2. This phase involves three algorithms:

1. Algo 1 identifies the relevant documents for a given topic.
2. Algo 2 detects less similar documents in the relevant set of documents.
3. Algo 3 ascertains whether the new annotated document with a topic is relevant and less similar to a sub set of relevant and less similar documents of this topic.

First, the most relevant documents of each topic t_i are selected. For each document of a topic t_i, Algo 1 checks whether its most important terms are the same as the most important terms of the topic t_i. To identify the most important terms of a given document D_j, the tf-idf of each term W_i in the corpus C_{ti} is computed using equation (1):

$$f(W_i, D_j, C_{ti}) = TF - IDF(W_i, D_j, C_{ti}) = TF(W_i, D_j) * \log_2\left(\frac{|C_{ti}| - M_i}{IDF(W_i, C_{ti})}\right) \quad (1)$$

where $TF(W_i, D_j)$, $IDF(W_i, C_{ti})$ and M_i denote the number of occurrences of W_i in document D_j, the number of documents in the corpus C_{ti} where W_i appears, and the number of documents in the corpus C_{ti}, respectively.

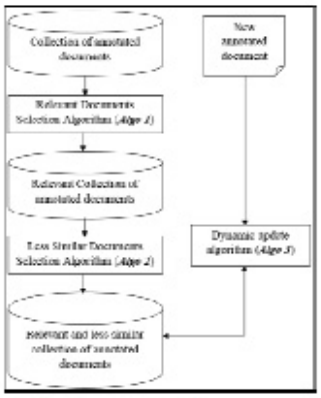


Figure 2. Relevant and less similar document selection process phase – Architecture overview

Equation (1) allows SATD to find, for each document D_j, the vector V_{Dj}= { $(W_a, f(W_a,D_j,C_{ti}))$, …, $(W_i, f(W_i,D_j,C_{ti}))$,…, $(W_{|Dj|}, f(W_{|Dj|},D_j,C_{ti}))$} where in the couple $(W_i, f(W_i,D_j,C_{ti}))$, W_i denotes a term and $f(W_i,D_j,C_{ti}))$ its tf-idf in the whole corpus C_{ti}. To identify the most important terms of a given topic t_i, the tf-itf of each term W_k that appears at least one time in at least one document of corpus C_{ti} is computed with formula (2):

$$g(W_k,t_i) = TF - ITF(W_k,t_i) = TF(W_k,t_i) * \log\left(\frac{|T| = n}{ITF(W_k)}\right) \quad (2)$$

where $TF(W_k,t_i)$, $ITF(W_k)$ and |T| denote the number of occurrences of W_k in all the documents of corpus C_{ti}, the number of topics where W_k appears, and the number of topic, respectively.

Equation (2) allows SATD to find, for each topic ti, the vector Vti= { (W1 , g(W1,ti)), …, (Wk, g(Wk,ti)) ,…, (WNi , g(WNi,ti))} where in the couple (Wk , g(Wk,ti)), Wk denotes a term and g(Wk,ti) its tf-itf in the whole corpus T.

Let N_i be the number of terms of the vocabulary of C_{ti} and N_{Dj} = |Dj| be the number of terms of the vocabulary of D_j. In this context, N_i is larger than N_{Dj}. To determine the number of terms to consider the document relevant, SATD computes the standard deviation σ and the average *avg* of the number of distinct terms in the documents for the topics. SATD uses the standard deviation. The standard deviation σ_{ti} of topic t_i is given by equation (3):

$$\sigma_{ti} = \sqrt{\frac{\sum_{j=1}^{|C_{ti}|=M_i}(|D_j| - avg_{ti})^2}{|C_{ti}| = M_i}} \quad (3)$$

where the average number of terms avg_{ti} of topic t_i is computed using equation (4).

$$avg_{ti} = \frac{\sum_{j=1}^{|C_{ti}|=M_i}|D_j|}{|C_{ti}| = M_i} \quad (4)$$

Next, to compute the number of distinct terms to consider, SATD uses equation (5).

$$E_{ti} = avg_{ti} - \sigma_{ti} \quad (5)$$

The score for each document D_j in the topic t_i is computed next:

1. SATD sorts, for each document D_j of corpus C_{ti}, the vector V_{Dj} by $f(W_i,D_j,C_{ti})$ in descending order.
2. SATD computes the BMscore of D_j using equation (6):

$$BMscore\{D_j\} = \sum_{|E_i|}^{1} g(W_i,t_i) \quad (6)$$

where $\sum_{|E_i|}^{1}$ are the first $|E_i|$ terms W_i of D_j with the highest value of $f(W_i,D_j,C_{ti})$ in the whole corpus C_{ti}.

In order terms, BMscore is the summation of the tf-itf in the whole corpus C of the first $|E_i|$ terms W_i of D_j with the highest tf-idf in the whole corpus C_{ti}. Finally, based on the BMscore of each document D_j of corpus C_{ti}, SATD selects the most relevant documents of corpus C_{ti}. SATD obtains the sub-corpus C'_{ti} of the most relevant documents using equation (7):

$$C_{ti} = \left[C'_{ti} = \bigcup_{\alpha}\{D_k\}\right] \cup \left[\bigcup_{|M_i|-\alpha}\{D_f\}\right] \quad (7)$$

where BMscore (Dk) > BMscore (Dj).

Note that α is a threshold determined by empirical experimentation based on the particular document collection. $C_{ti}^{t} = \{D_{k_1}, \ldots, D_{k_i}, \ldots, D_{k_n}\}$ is obtained where $M_t \geq M_t^i = \alpha$. Algorithm 1 of appendix A explains, in detail, the selection process of relevant documents for a given topic.
The less similar documents of sub-corpus C_{ti}^{t} for the topic t_i are then selected. SATD defines a similarity threshold β by empirical experimentation based on the particular document collection where C_{ti}^{lt} is the sub-corpus of C_{ti}^{t} that contains the less similar documents.

SATD sorts the documents of C_{ti}^{t} according to their BMscore. SATD first puts the document with the largest BMscore in C_{ti}^{lt}; then, based on the order of largest BMscore, SATD compares the semantic similarity of each element of C_{ti}^{t} with the rest of element of $C_{ti}^{t}C_{ti}^{t}$. If no document of C_{ti}^{t} is semantically similar to a given document of C_{ti}^{t}, this given document is added to C_{ti}^{lt}. When the semantic similarity between two documents is less than or equal to β, SATD assumes they are not similar. Finally, when a new document annotated with topic t_i, is added to the corpus C_{ti}, SATD computes its BMscore in order to ascertain whether this new document must be added to C_{ti}^{lt} or not.

For example, let IDF_{ti}^{s} be the idf vector of the vocabulary of corpus C_{ti} at state *s* and ITF^{s} be the itf vector of the vocabulary of corpus C at state *s*. The state is the situation of the collection before adding the new document:

$IDF_{ti}^{s} = (IDF(W_1, C_{ti}), \ldots, IDF(W_k, C_{ti}), \ldots, IDF(W_{Nt}, C_{ti}))$ and

$ITF^{s} = (ITF(W_1), \ldots, ITF(W_k), \ldots, ITF(W_{Nt}))$. Let TF_{ti}^{s} be the tf vector of the vocabulary of corpus C_{ti} at the state *s*:

$$TF_{ti}^{s} = (TF(W_1, t_i), \ldots, TF(W_k, t_i), \ldots, TF(W_{Nt}, t_i)).$$

Based on vector IDF_{ti}^{s}, SATD computes the TF-IDF of each term *W* of *d* of each term w of *d* using Equation (8):

$$f(W, d, C_{ti}) = TF - IDF(W, d, C_{ti}) = TF(W, d) * \log\left(\frac{|C_{ti}|}{IDF(W, C_{ti}) + 1}\right) \quad (8)$$

Next, SATD ranks the vocabulary of d according to their $f(W, d, C_{ti})$ and selects the E_{di} terms *W* of d with highest $f(W, d, C_{ti})$. Based on the vectors ITF_{ti}^{s} and TF_{ti}^{s}, SATD computes the TF-ITF of each selected term *W* of *d* using equation (9):

$$g(W, t_i) = TF - ITF(W, t_i) = [TF(W, t_i) + TF(W, d)] * \log\left(\frac{|T|}{ITF(W_k)}\right) \quad (9)$$

SATD obtains the BMscore(d) of new document *d* by summation of the $g(W, t_i)$ term. If BMscore(d) is greater than the smallest BMscore of C_{ti}^{t} document, SATD uses Algorithm 2 to make a semantic similarity computation and then performs an update of C_{ti}^{lt} if necessary.

3.4.2. Not annotated documents semantic term graph generation - process phase

The semantic term graph allows one to convert a set of lines of terms into a graph by extracting semantic and co-occurrence relations between terms. To generate the semantic term graph BM-SemGraph: (1) first the co-occurrence clusters are generated and then optimized, (2) after optimization, the key terms and links between the clusters are extracted and (3) finally, the semantic topic is generated and semantic term graph extracted.

The BM-SemGraph has one node for each term in the vocabulary of the document. Edges in a BM-SemGraph represent the co-occurrence of the corresponding keywords and are weighted by the count of the co-occurrences. Note that, in contrast to existing graph-based approaches, the co-occurrence between A and B is different from the co-occurrence between B and A. This difference allows one to retain the semantic sense of co-occurrence terms. Figure 3 presents an overview of the architecture of

the semantic term graph generation process phase. The term graph process and BM-SemTopic process generate the semantic graph in order to enrich the term graph with semantic information; indeed, the terms graph and semantic graph are merged to provide Semantic term graph, called BM-SemGraph.

The term graph process consists of three steps: (1) Co-occurrence clusters generation, (2) Clusters optimization and (3) Key terms extraction. The BM-SemTopic process consists of two steps: (1) Semantic topic generation and (2) Semantic graph extraction.

Step 1: Co-occurrence clusters generation

For the co-occurrence graph, the assumption is that terms that have a close relation to each other may be linked by the co-occurrence link. The relation between two terms W_i and W_j is measured by their conditional probability. Let D be a document and $V_D = (w_1, w_2; \ldots, w_N)$ be the terms of D and L_D be the number of lines of D.

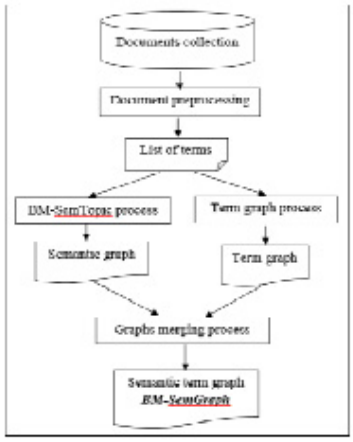


Figure 3. New document semantic term graph process phase - Architecture overview

The conditional probability $p(\overrightarrow{W_i, W_j}^{\varepsilon})$ of $\overrightarrow{W_i, W_j}^{\varepsilon}$ is computed using equation (10) where ε (determined by experimentation) denotes the minimum distance between W_i and W_j and the distance between two terms is the number of terms that appear between them for a given line.

$$p\left(\overrightarrow{W_i, W_j}^{\varepsilon}\right) = \sum_{l=1}^{L_D} \frac{N^{line\ l}\left(\overrightarrow{W_i, W_j}^{\varepsilon}\right)}{\left\lceil \frac{N(line\ l)}{\varepsilon} \right\rceil} \tag{10}$$

where $N^{line\ l}\left(\overrightarrow{W_i, W_j}^{\varepsilon}\right)$ denotes the number of times that W_i and W_j co-occur with a minimum distance ε and where W_i appears before W_j, and *N(line l)* denotes the number of terms of the line l.
To formally define a relation between two terms W_i and W_j, their frequent co-occurrence measured by the conditional probability $p(\overrightarrow{W_i, W_j}^{\varepsilon})$, needs to exceed the co-occurrence threshold. The co-occurrence threshold is also determined by experimentation. Note that frequent co-occurrence is oriented. This allows one to retain the semantic orientation of the links between terms.

Next, the oriented links are transformed into simple links without losing the semantic context. To perform this transformation, three rules are applied - see Figure 4.

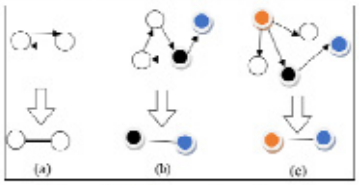


Figure 4. Link transformation rules

In Figure 4a, two nodes with two oriented links are transformed into one simple link. In this case, this type of link cannot be pruned and its weight is given by equation (11):

$$w(W_i, W_j) = p\left(\overrightarrow{W_i, W_j}^{\alpha}\right) + p\left(\overrightarrow{W_j, W_i}^{\alpha}\right) \tag{11}$$

In Figure 4b, where several nodes are linked by oriented links and there is an oriented path to join each of them, only the nodes with a link to other nodes not in the oriented path are retained. The black node becomes the representative of the other nodes.

In Figure 4c, where one node A is linked to several nodes and the links are oriented from A towards the other nodes, node A becomes the representative of the other nodes and the other nodes are removed. This is the case for the red node where the link between the black node and blue node is removed and a new link is added between the red node and the blue node. Let G be a set of nodes where W_i is the representative node. Let G' be the sub set of G which are linked to a node W_j not in G. Figure 5 illustrates G and G'. The weight of the link between W_i and W_j is given by equation (12):

$$w(W_i, W_j) = \sum_{W_k \in G'} p\left(\overrightarrow{W_k, W_j}^{\alpha}\right) + p\left(\overrightarrow{W_j, W_k}^{\alpha}\right) \tag{12}$$

Equation (12) is applied in the case of Figure 4b and 4c to compute the weight of the link between a representative node and another node. Finally, the rest of the oriented links are transformed into simple links and their weights computed using equation (11).

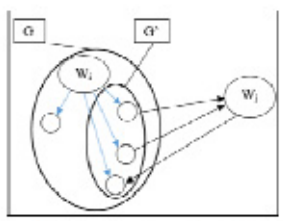


Figure 5. Representation of the computation of weight after removing some nodes

Step 2: Cluster optimization

To enhance quality, clusters should be pruned, such as by removing weak links or partitioning sparse cluster into cohesive sub-clusters. Clusters are pruned according to their connectedness. The link e is pruned when no path connects the two ends of e after it is pruned. As shown in Figure 6, the link between the black node and the green node should be pruned.

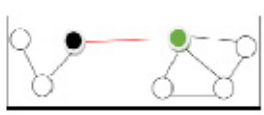


Figure 6. Clusters optimization

Secondly, cliques are identified. In graph theory, a clique is a set of nodes which are adjacent pairs (?) (or a two-by-two set of nodes?) as shown in Figure 7.

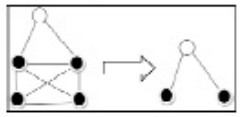


Figure 7. Clique reduction

Let C be the clique and W_i and W_j be the nodes of C that are linked to another node. The weight between W_i and W_j is given by equation (13):

$$w(W_i, W_j) = \underset{\substack{W_k \in C \\ W_k \neq i}}{\mathrm{MAX}}[w(W_k, W_i)] \tag{13}$$

Step 3: Key term extraction

To extract key terms, the relation between a term and a cluster is measured. It is assumed that the weight of a term in a given cluster may be used to determine the importance of this term for the cluster. Let R be the set of nodes of the cluster C where the node W_i is inside. The weight of W_i in the cluster C is given by equation (14):

$$f(W_i) = \sum_{W_j \in R} w(W_i, W_j) \tag{14}$$

To identify a term as a key term, a sort of terms is performed based on their weights regardless of the clusters that they are in. Next, the NumKeyTerm terms that have the largest weights are selected as Key Terms. NumKeyTerm is a parameter.

Step 4: Semantic topic generation

Semantic topic generation combines a correlated topic model (CTM) [21] and a domain knowledge model (DKM) [22], called BM semantic topic model (BM-SemTopic), to build the real semantic topic model. In LDA, a topic is a probability distribution over a vocabulary. It describes the relative frequency each word is used in a topic. Each document is regarded as a mixture of multiple topics and is characterized by a probability distribution over the topics.

A limitation of LDA is its inability to model topic correlation. This limitation stems from the use of the Dirichlet distribution to model the variability among topic proportions. In addition, standard LDA does not consider domain knowledge in topic modeling. To overcome these limitations, BM-SemTopic combines two models: (1) A correlated topic model (CTM) [21] that makes use of a logistic normal distribution and (2) A domain knowledge model (DKM) [22] that makes use of the Dirichlet distribution.

BM-SemTopic uses a weighted sum of CTM and DKM to compute the probability distribution of term W_i on the topic z. The sum is defined by equation (15):

$$h(W_i|z) = \omega CTM(W_i|z) + (1 - \omega)\, DKM(W_i|z) \tag{15}$$

where ω is used to give more influence to one model based on the term distribution of topics.

When the majority of terms are located in a few topics, this means the domain knowledge is important and ω must be small. BM-SemTopic develops the CTM where the topic proportions exhibit a correlation with the logistic normal distribution and incorporates the DKM. A key advantage of BM-SemTopic is that it explicitly models the dependence and independence structure among topics and words, which is conducive to the discovery of meaningful topics and topic relations.

CTM is based on a logistic normal distribution. The logistic normal is a distribution on the simplex that allows for a general pattern of variability between the components by transforming a multivariate normal random variable. This process is identical to the generative process of LDA except that the topic proportions are drawn from a logistic normal distribution rather than a Dirichlet distribution. The

strong independence assumption imposed by the Dirichlet in LDA is not realistic when analyzing document collections where one may find strong correlations between topics. To model such correlations, the covariance matrix of the logistic normal distribution in the BM-SemTopic correlated topic model is introduced.

DKM is an approach to incorporation of such domain knowledge into LDA. To express knowledge in an ontology, BM-SemTopic uses two primitives on word pairs: Links and Not-Links. BM-SemTopic replaces the Dirichlet prior by the Dirichlet Forest prior in the LDA model. Then, BM-SemTopic sorts the terms for every topic in descending order according to the probability distribution of the topic terms. Next it picks up the high-probability terms as the feature terms. For each topic, the terms with probabilities higher than half of the maximum probability distribution are picked up (experiment indicates it is non-sensitive on this parameter).

Step 5: Semantic term graph extraction

To enrich the term graph, the semantic topic needs to be converted into a semantic graph that consists of semantic relations between the semantic terms. To discover these relations, the semantic aspect is included making use of WordNet::Similarity [23]. Based on the structure and content of the lexical database WordNet, WordNet::Similarity implements six measures of similarity and three measures of relatedness. Measures of similarity use information found in a hierarchy of concepts (or synsets) that quantify how much concept A is like (or is similar to) concept B.

First, each generated feature term at step 4 is the candidate for a semantic term where it is assumed the other terms represent the vocabulary associated with the semantic topic. In Figure 8a, the blue node denotes the feature terms of each semantic topic. Next, duplicate terms from the candidates are removed. If there is more than one topic that has the same term W_j in the semantic term candidate, only the topic z with the highest term probability distribution $h(W_j|z)$ is retained W_j as the semantic term candidate. It follows then that following this step the semantic term candidates of different topics are exclusive to each other. Figure 8b shows the remaining candidates by semantic topic.

To remove similar terms, the measure path (one measure of similarity of WordNet::Similarity [23]) is used to evaluate similarity between two terms. The measure path of WordNet::Similarity is a baseline that is equal to the inverse of the shortest path between two concepts. When the semantic term candidates of different topics are identified, the semantic value of each topic's candidates is computed. The semantic value of each term W_i, is given by equation (16):

$$SEM(W_i|z) = TP - ITP(W_i|z) = h(W_i|z) * \log\left(\frac{|z|}{\sum_{z \in Z} h(W_i|z)}\right) \quad (16)$$

where Z denotes the set of semantic topics. TP-ITP is inspired by the tf-idf formula, where TP is term probability and ITP inverse topic probability.

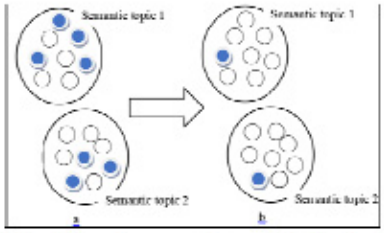


Figure 8. Candidates for semantic term identification (a and b)

Semantic links between semantic terms for the term graph are constructed using the vector measure, one of the measures of relatedness of WordNet::Similarity [23]. The vector measure creates a co–occurrence matrix for each word used in WordNet glosses from a given corpus, and then represents each gloss/concept with a vector that is the average of these co–occurrence vectors.

Let W_i and W_j be semantic terms of the synsets A and B, respectively. Let $\vec{A} = (a_1, \ldots, a_n)$ and $\vec{B} = (b_1, \ldots, b_n)$ be the co–occurrence vectors of A and B, respectively. Let V_z be the set of semantic terms of the semantic topic Z. The weight of the link between W_i and W_j is computed by equation (17):

$$Dis(W_i, W_j \mid z) = \frac{SEM(W_i|z) + SEM(W_j|z)}{\sum_{W_k \in V_z} SEM(W_k|z)} \times \sqrt{\sum_{i=1}^{n} (a_i - b_i)^2} \quad (17)$$

To discover a semantic relation between two terms, the semantic distance is computed. The semantic distance between two terms is the shortest path between the terms using equation (18):

$$SEMDis(W_i, W_j \mid z) = \underset{pa \in P}{\mathrm{MIN}} \left[\sum_{W_k \in pa} Dis(W_i, W_k \mid z) \right] \quad (18)$$

where pa, W_k, and P denote a path between W_i and W_j in the thesaurus, a term on a path pa and the set of paths pa between W_i and W_j, respectively.

To formally define a semantic relation between two terms W_i and W_j, the semantic distance $SEMDis(W_i, W_j \mid z)$ must not exceed the semantic threshold. The semantic threshold is determined by experimentation.

The last process to generate the semantic term graph BM-SemGraph is a merging of the term graph and the semantic graph. The term graph and semantic graph are merged by coupling the co-occurrence relation and the semantic relation. New terms are added as semantic terms and new links are added as semantic links if they do not appear in the term graph. For each link between two nodes W_j and W_k of the merged graph, the weight, called the BM Weight (BMW), for a given topic t_i is computed using equation (19):

$$BMW(W_j, W_k \mid t_i) = \frac{\lambda}{SEMDis(W_j, W_k \mid t_i)} + (1 - \lambda) \times w(W_i, W_j) \quad (19)$$

where λ determined by experimentation.

In order to optimize the clusters of BM-SemGraph, the weak links or partitioning of sparse clusters are removed. At this step, each cluster is considered a topic and the terms of the cluster become the terms of the topic.

3.4.3. Topic detection - process phase

Figure 9 presents the process used by SATD to assign topics to a document.

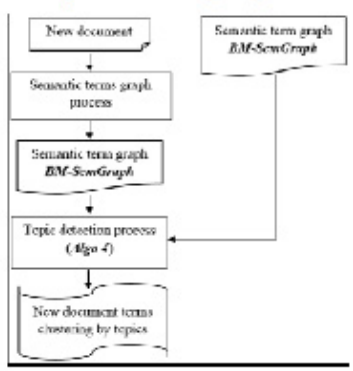


Figure 9. Topic detection process phase - Architecture overview

Topics that may be associated with a new document are detected based on the BM-SemGraph. Note that the BM-SemGraph is obtained using a collection of documents. In this case, the likelihood of detecting topics among a collection of documents is high and must be computed. To accomplish this, the feature vector of each topic based on the clusters of BM-SemGraph is computed. The feature vector of a topic is calculated using the BMRank of each topic term. Let A be the set of nodes of BM-SemGraph directly linked to term W_j in the topic t_i. The score for the term W_j is given by equation (20):

$$BMRank(W_j|t_i) = \frac{\sum_{W_k \in A} BMW(W_j, W_k \mid t_i)}{|A|} \tag{20}$$

The term with the largest BMRank is called the main term of the topic; other terms are secondary terms. The same processes are used to obtain the BM-SemGraph of an individual document *d* and the feature vectors of topics t_j^d. Next, the similarity between each topic t_i and the topics t_j^d of document *d* is computed in order to detect document topics. Let W_i be a master term of topics t_j^d and a master or secondary term of t_i, B be the intersection of the set of terms of BM-SemGraph directly linked to term W_j in the cluster of topic t_i and the set of terms of BM-SemGraph of individual document *d* directly linked to term W_j in the cluster of topic t_j^d, and C be the union of the set of terms of BM-SemGraph directly linked to term W_j in the cluster of topic t_i and the set of terms of BM-SemGraph of individual document *d* directly linked to term W_j in the cluster of topic t_j^d. The similarity between t_i and topic t_j^d is computed with equation (21):

$$Sim(t_i|t_j^d) = \frac{\sqrt{\sum_{W_k \in B} \left(BMW(W_j, W_k \mid t_i) - BMW(W_j, W_k \mid t_j^d)\right)^2}}{\sqrt{\sum_{W_k \in C} \left(BMW(W_j, W_k \mid t_i) - BMW(W_j, W_k \mid t_j^d)\right)^2}} \tag{21}$$

Here, t_i and topic t_j^d are considered to be similar when their similarity $Sim(t_i|t_j^d)$ does not exceed the vector similarity threshold. Finally, the document d is assigned to topics that are similar to its feature vectors.

3.4.4. Training - process phase

The training process establishes a terms graph based on the relevant and less similar documents for a given topic t_i. To form the terms graph for a given topic, preprocessing of its relevant and less similar documents is first carried out, a set of lines is obtained where each line is a list of terms, and the co-occurrence of these terms is then computed. Let Doc be a document and $V_{Doc} = (w_1, w_2; \ldots, w_N)$ be the terms of Doc. The co-occurrence of $co\left(\overline{W_i, W_j}^{\varepsilon}\right)$ of W*i* and W_j where ε denotes the minimum distance between W_i and W_j is computed using equation (22):

$$co\left(\overline{W_i, W_j}^{\varepsilon}\right) = \sum_{l=1}^{L_{Doc}} \frac{N^{line\ l}\left(\overline{W_i, W_j}^{\varepsilon}\right)}{\left\lceil \frac{N(line\ l)}{\varepsilon} \right\rceil} \tag{22}$$

where $N^{line\ l}\left(\overline{W_i, W_j}^{\varepsilon}\right)$ denotes the number of times that W_i and W_j co-occur with a minimum distance ε, regardless of the order of appearance, and N(line l) denotes the number of terms of line l. A relation between two terms W_i and W_j is formally defined when the computed co-occurrence between them exceeds the co-occurrence threshold determined by experimentation. Figure 10 presents an overview of the training process phase.

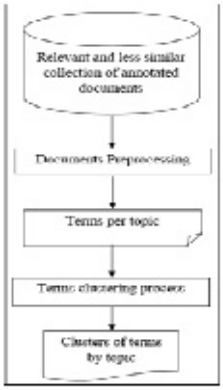


Figure 10. Training process phase - Architecture overview

3.4.5 Topics refining - process phase

Figure 11 presents the process used by SATD to refine the detected topics making use of relevant documents already annotated by humans based on existing or known topics. Following this process, three lists of topics are obtained: a list of new topics, a list of similar existing topics and a list of not similar existing topics. The list of existing topics that match new document detected topics is identified based on the new document detected topics and annotated documents by topic (existing topics). The clusters of terms by topic are identified based on the collection of relevant and less similar documents. Note: each topic is a cluster of terms graph. Therefore, a graph matching technique is a good candidate to perform topic similarity detection. Next, using our graph matching technique, the clusters of terms by topics of relevant and less similar collection of annotated documents which match with CTG are identified, for each cluster of terms graph by topic (CTG) of the new document.

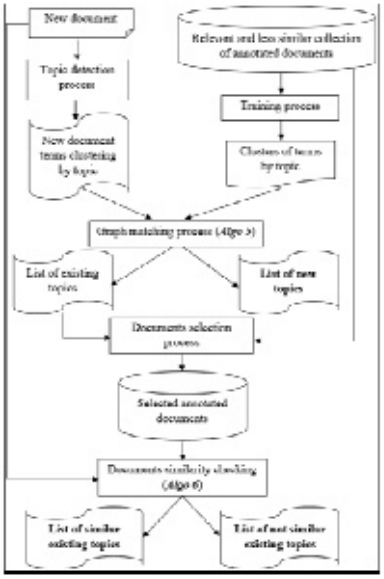


Figure 11. Topic refining process phase - Architecture overview

The matching score between two clusters is then computed. Let H be the new document terms graph and G be the terms graph obtained by a training process applied on the collection of relevant and less similar documents annotated by topics, C_j^d be a cluster of H associated to topic t_j^d and C_i be a cluster of G associated with topic t_i, and W_i and W_j be two terms of cluster C_j^d; the link matching function $g(\overline{W_iW_j})$ between W_i and W_j is defined by equation (23):

$$g: C_j^d \times C_j^d \rightarrow IR$$
$$g(\overline{W_iW_j}) = \begin{cases} \text{Minhop Cluster Path}(W_i,W_j) & \text{if path between } W_i,W_j \\ \text{1+Maxhop Cluster Path} & \text{if no path between } W_i,W_j \end{cases} \quad (23)$$

For a direct link $\overline{W_iW_j}$ (only one hop between W_i and W_j) of cluster C_j^d, the process checks whether there is a path between W_i and W_j in the cluster C_i, regardless of the number of hops:

1. If paths exist between W_i and W_j in the cluster C_i, $g(\overline{W_iW_j})$ is the number of hops of the shortest path between W_i and W_j, in term of hops.
2. Otherwise, $g(\overline{W_iW_j})$ is the number of hops of the longest path that exists in the cluster C_i incremented by 1.

Using the link matching function, the matching score between two clusters C_i^d and C_i is given by equation (24):

$$o: H \times G \rightarrow]0;1]$$
$$o(C_j^d, C_i) = \frac{|C_j^d|}{\sum_{W_i,W_j \in C_j^d} g(\overline{W_iW_j})} \quad (24)$$

where $|C_j^d|$ is the number of links in clusters C_i^d. For a better understanding, consider the term graphs in Figure 12.

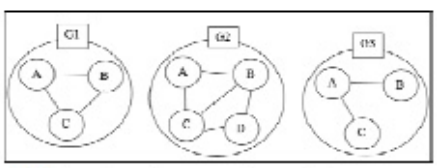


Figure 12. Illustration of term graphs matching score computation

According to Figure 12, o(G1,G2) = 3/3 = 1 while o(G2,G1) = 5/9 and o(G1,G3) = 3/5 while o(G3,G1) = 2/2 = 1. The clusters of H and G whose matching scores exceed a term cluster matching threshold are considered as matching and are assumed to be the same topics. Otherwise, the clusters of H that do not match any clusters of G, are assumed to be new topics. Note that the term cluster matching threshold is determined by experimentation. Based on the H and G clusters that match, the relevant and less similar documents per existing topic that may have the same topic as the new document are identified. Making use of this set of selected documents, the similarity between the new document and each relevant and less similar document of each existing topic i is measured. Let D be the union of the new document *d* and a set of relevant and less similar documents of existing topics t_i that are selected by documents selection and $W = \{W_1, \ldots, W_m\}$ the set of distinct terms occurring in D. The defined m-dimensional vector represents each document of D. For each term of W, its tf-idf is computed using equation (1). This allows one to obtain the vector $\overrightarrow{t_d} = (\text{tfidf}(W_1, d, t_i), \ldots, \text{tfidf}(W_m, d, t_i))$. When documents are represented as term vectors, the similarity of two documents corresponds to the correlation between the vectors. Here, cosine similarity is applied to measure this similarity. The cosine similarity is defined as the cosine of the angle between vectors. An important property of the cosine similarity is its independence of document length. Given two documents $\overrightarrow{t_{d1}}$ and $\overrightarrow{t_{d2}}$, their cosine similarity is computed using equation (25):

$$SimCos(\overrightarrow{t_{d1}}, \overrightarrow{t_{d2}}) = \frac{\overrightarrow{t_{d1}} \cdot \overrightarrow{t_{d2}}}{|\overrightarrow{t_{d1}}| \times |\overrightarrow{t_{d2}}|} \quad (25)$$

Note that it is already assumed that when the similarity $SimCos(\overrightarrow{t_{d1}}, \overrightarrow{t_{d2}})$ of two documents d1 and d2 is less than the similarity threshold β, the documents are not similar. The computation of document similarity allows SATD to classify the existing topics into: (1) Similar existing topics and (2) Not similar existing topics.

4. EVALUATION USING SIMULATIONS

This section presents an evaluation of SATD performance using simulations. To perform these simulations, an experimental environment called Libër was used. Libër was developed to provide a simulator to prototype SATD algorithm.

4.1. Dataset and parameters

To evaluate SATD, real datasets from different projects that have digital and physical library catalogues were used. These datasets, consisting of 25,000 documents with a vocabulary of 375,000 words, were selected using average TF-IDF for the analysis. The documents covered 20 topics. The number of documents per topic or emotion was approximately equal. The average number of topics per document was 7 while the average rating emotion number per document was 4. 15,000 documents of the dataset were used for the training phase and the remaining 100 used for the test. Note that the 10,000 documents used for the tests were those that had more annotated topics or a higher rating over emotions.

To measure the performance of topic detection, comparison of detected topics with annotation topics were carried out. Table 2 presents the values of the parameters used in the simulations. The server characteristics for the simulations were: Dell Inc. PowerEdge R630 with 96 Ghz (4 x Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz, 10 core and 20 threads per CPU and 256 GB memory running VMWare ESXi 6.0.

Table 2. Simulation parameters

Parameter	Value	Parameter	Value
ε	3	α	100
NumKeyTerm	8	co-occurrence threshold	0.75
ω	0.5	semantic threshold	1
β	0.7	term cluster matching threshold	0.45
λ	0.6		

4.2. Performance criteria

SATD performance was measured in terms of running time [8] and accuracy [15] [14]. Note that in the library domain, the most important criteria was precision while resource consumption was important for the software providers.

The running time, denoted by Rt, was computed as follows:

$$Rt = Et - Bt$$

where *Et* and denotes the time when processing is completed and *Bt* the time when it started. To compute the accuracy, let $T_{annotated}$ and $T_{detected}$ be the set of annotated topic and the set of detected topics by SATD for a given document d. The accuracy of topics detection, denoted by A_d^t, was computed as follows:

$$A_d^t = \frac{2.|T_{annotated} \cap T_{detected}|}{|T_{annotated}| + |T_{detected}|}$$

Simulation results were averaged over multiple runs with different pseudorandom number generator seeds. The average accuracy, *Ave_acc*, of multiple runs was given by:

$$Ave\_acc = \frac{\sum_{i=1}^{I}\left(\frac{\sum_{d \in TD} A_d^T}{|TD|}\right)}{I}$$

where TD denotes the number of tests documents and I denotes the number of test iterations. The average running time, *Ave_run_time,* was given by:

$$Ave\_run\_time = \frac{\sum_{i=1}^{I} Rt}{I}$$

4.3 Comparison approaches

SATD performance was evaluated in terms of running time and accuracy. The dataset and parameters mentioned above were applied. SATD performance was compared to the approaches described in [15], [14], [4] and [8], referred to as LDA-IG (probabilistic and graph approach), KeyGraph (graph analytical approach), LDA (probabilistic approach) and HLTM, respectively. LDA-IG, KeyGraph, LDA and HLTM were selected because they are text-based and long text approaches. Table 3 presents the characteristics of the comparison approaches. Our prototype approach SATD is the only one that is really semantic and takes into account the correlated topic and domain knowledge.

Table 3. Topic detection approaches for comparison

Approach	Granularity	Description	Training phase	Refining	Semantic	Topic corre-lation	Domain knowledge
LDA-IG [15]	D	P,G	Yes	No	No	No	No
KeyGraph [14]	D	G	Yes	No	No	No	No
LDA [4]	D	P	No	No	No	No	No
HLTM [8]	D	P,G	Yes	No	No	No	No
SATD	C	S,P,G	Yes	Yes	Yes	Yes	Yes

D: document; C: Configurable as desired; P: Probabilistic based; G: Graph based; S: Semantic based.

4.4. Results analysis

Figure 13 presents the average running time of the detection phase when the number of documents used for the tests were varied. Training times were excluded as this phase was performed only one time. However, the SATD training phase required more time than the other approaches. This was justified by the fact that SATD identifies the relevant and less similar documents used for training phase. Figure 13 also shows that the average running time increased with the number of test documents. Indeed, the bigger the number of test documents, the longer the time to perform detection and, ultimately, the higher the average running time.

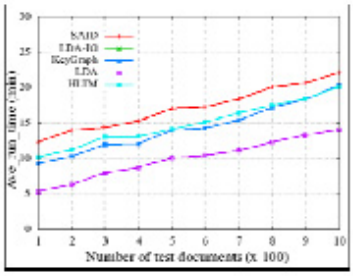


Figure 13. Topic detection - Average running time versus number of documents for test phase

It was also observed that LDA outperforms the other approaches. LDA produced an average of 1.37 sec per document whereas SATD produced an average of 2.62 sec per document. The average relative improvement (defined as [Aver._runtime of SATD – Aver._runtime of LDA]) of LDA compared with

SATD was approximately 1.25 sec per document. The short run times of LDA were due to the fact that LDA did not perform a graph treatment. Graph processing algorithms are very time consuming. Other approaches also outperformed SATD on the running time criteria since SATD performed topic refining in order to increase accuracy.

Figure 14 shows the average accuracy when varying the number of detected topics. For the five approaches, the average accuracy decreased with the number of detected topics. The increase in the number of subjects to detect led to decreased accuracy. However, in terms of accuracy, SATD outperformed the approaches used for comparison. SATD produced an average accuracy of 79.50% per topic while LDA-IG, the best among the approaches used for comparison, produced an average of 61.01% per topic. The average relative improvement in accuracy (defined as [Ave_acc of SATD – Ave_acc of LDA-IG]) of SATD compared to LDA-IG was 18.49% per topic. The performance of SATD is explained as follows: (1) SATD used the relevant documents for training phase, (2) SATD refined its detection topic results by measuring new document similarity with relevant and less similar annotated documents, and (3) SATD combined correlated topic model and domain knowledge model instead of LDA.

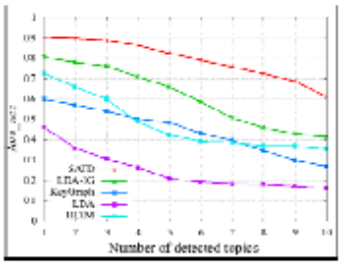


Figure 14. Accuracy for number of detected topics for 5 comparison approaches

Figure 14 also shows that SATD produced an average accuracy of 90.32% for one detected topic and 61.27% for ten detected topics compared to 80.29% and 41.01% respectively for LDA-IG. The gap between SATD accuracy and LDA-IG accuracy was 10.03% for one detected topic and 20.26% for ten detected topics. This meant that SATD was by in large more accurate than LDA-IG in detecting several topics.

The Figure 15 presents the average accuracy when varying the number of training documents of the learning phase. LDA was not included in the scenario since no training phase was performed. Figure 15 shows that the average accuracy increased with the number of training documents. The larger the number of training documents, the better the knowledge about word distribution and co-occurrence and, ultimately, the higher the detection accuracy. However, the accuracy remained largely stable for very high numbers of training documents. When the number of documents of a collection was larger, the number of vocabulary words remained constant, and the term graph did not change. It also shows that HLTM was the approach whose detection accuracy was the first to reach stability at 10,000 training documents. HLTM builds a tree instead of a graph as the other approaches and its tree has less internal roots to identify topics. However, SATD and LDA-IG outperformed HLTM in terms of accuracy.

Figure 15 also shows that SATD outperformed LDA-IG on the accuracy criteria. For example, SATD demonstrated an average accuracy of 73.49% per 2,000 training documents while LDA-IG produced an average accuracy of 50.86% per 2,000 training documents. The average relative improvement of SATD compared to LDA-IG was 22.63% per 2,000 training documents. The better performance of SATD followed from its use of a specific domain knowledge model. SATD did not require a large number of documents for the training phase.

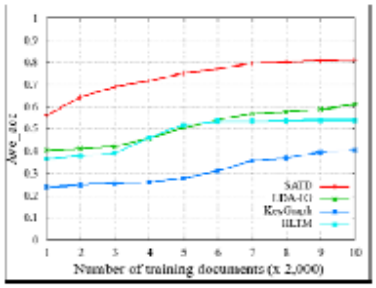


Figure 15. Topic detection - accuracy for number of training documents

In conclusion, the 1.25 sec running time per document increase was a small price to pay for the larger average accuracy of topic detection (18.49%).

5. SUMMARY AND FUTURE WORK

The goal of this paper was to increase the findability (search engines) of user interests using semantic metadata enrichment model and algorithm. Words themselves have a wide variety of definitions and interpretations and are often utilized inconsistently. While topics may have no relationship to individual words, thesauri express associative relationships between words, ontologies, entities and a multitude of relationships represented as triplets. This paper presented an enhanced implementation of SMESE [1] model using SATD engine for topic metadata enrichments.

To help users find interest-based contents, this paper proposes to enhance the SMESE platform [1] through text analysis approaches for topic detection. This paper presents the design, implementation and evaluation of the algorithm SATD focusing on semantic topic extraction. The SATD topic metadata enrichments prototype allows to: (1) generate semantic topics by text, and multimedia content analysis using the proposed SATD (Scalable Annotation-based Topic Detection) algorithm and (2) implement rule-based semantic metadata internal enrichment. Table 1 shows the comparison with most known text mining algorithms (e.g., AlchemyAPI, DBpedia, Wikimeta, Bitext, AIDA, TextRazor) and a new algorithm using keyword extraction, classification and concept extraction. It was noted that SATD algorithm support more attributes than the other algorithms evaluated.

In future work, the focus will be to generate learning-based literature review enrichment and abstract of abstract. It will assess each reference extracting topics to determine her ranking and her inclusion in the literature assistant review. One main goal is to reduce reading load by helping researcher to read only the most related selection of documents to literature review. Using text data mining, machine learning, and a classification model that learn from users annotated data and detected metadata the algorithms will assist the researcher to rank the relevant documents for his literature review for a specific topic and selection of metadata.

REFERENCES

[1] Brisebois R, Abran A, Nadembega A (2017) A Semantic Metadata Enrichment Software Ecosystem (SMESE) based on a Multi-platform Metadata Model for Digital Libraries. Accepted for publication in Journal of Software Engineering and Applications (JSEA) 10 (04)

[2] Salton G, Buckley C (1988) Term-weighting approaches in automatic text retrieval. Information Processing & Management 24 (5):513-523. doi:http://dx.doi.org/10.1016/0306-4573(88)90021-0

[3] Niu T, Zhu S, Pang L, El Saddik A (2016) Sentiment Analysis on Multi-View Social Data. Paper presented at the 22nd International Conference on MultiMedia Modeling (MMM), Miami, FL, USA, 4-6 Jan. 2016

[4] Blei DM, Ng AY, Jordan MI (2003) Latent Dirichlet Allocation. Journal of Machine Learning Research 3:993–1022

[5] Bijalwan V, Kumar V, Kumari P, Pascual J (2014) KNN based Machine Learning Approach for Text and Document Mining. International Journal of Database Theory and Application 7 (1):61-70. doi:http://dx.doi.org/10.14257/ijdta.2014.7.1.06

[6] Dumais ST (2004) Latent semantic analysis. Annual Review of Information Science and Technology 38 (1):188-230. doi:10.1002/aris.1440380105

[7] Cigarrán J, Castellanos Á, García-Serrano A (2016) A step forward for Topic Detection in Twitter: An FCA-based approach. Expert Systems with Applications 57:21-36. doi:http://dx.doi.org/10.1016/j.eswa.2016.03.011

[8] Chen P, Zhang NL, Liu T, Poon LKM, Chen Z (2016) Latent Tree Models for Hierarchical Topic Detection. arXiv preprint arXiv:160506650 [csCL]:1-44

[9] Moraes R, Valiati JF, Gavião Neto WP (2013) Document-level sentiment classification: An empirical comparison between SVM and ANN. Expert Systems with Applications 40 (2):621-633. doi:http://dx.doi.org/10.1016/j.eswa.2012.07.059

[10] Ghiassi M, Skinner J, Zimbra D (2013) Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network. Expert Systems with Applications 40 (16):6266-6282. doi:http://dx.doi.org/10.1016/j.eswa.2013.05.057

[11] Dang Q, Gao F, Zhou Y (2016) Early detection method for emerging topics based on dynamic bayesian networks in micro-blogging networks. Expert Systems with Applications 57:285-295. doi:http://dx.doi.org/10.1016/j.eswa.2016.03.050

[12] Cotelo JM, Cruz FL, Enríquez F, Troyano JA (2016) Tweet categorization by combining content and structural knowledge. Information Fusion 31:54-64. doi:http://dx.doi.org/10.1016/j.inffus.2016.01.002

[13] Hashimoto T, Kuboyama T, Chakraborty B (2015) Topic extraction from millions of tweets using singular value decomposition and feature selection. Paper presented at the 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), Hong Kong, China, 16-19 Dec. 2015

[14] Sayyadi H, Raschid L (2013) A Graph Analytical Approach for Topic Detection. ACM Transactions on Internet Technology 13 (2):1-23. doi:http://dx.doi.org/10.1145/2542214.2542215

[15] Zhang C, Wang H, Cao L, Wang W, Xu F (2016) A hybrid term–term relations analysis approach for topic detection. Knowledge-Based Systems 93:109-120. doi:http://dx.doi.org/10.1016/j.knosys.2015.11.006

[16] Bougiatiotis K, Giannakopoulos T (2016) Content Representation and Similarity of Movies based on Topic Extraction from Subtitles. Paper presented at the Proceedings of the 9th Hellenic Conference on Artificial Intelligence, Thessaloniki, Greece, 18-20 May 2016

[17] Salatino AA, Motta E (2016) Detection of Embryonic Research Topics by Analysing Semantic Topic Networks. Paper presented at the Semantics, Analytics, Visualisation: Enhancing Scholarly Data, Montreal, Quebec, Canada, 11 Apri. 2016

[18] Hurtado JL, Agarwal A, Zhu X (2016) Topic discovery and future trend forecasting for texts. Journal of Big Data 3 (1):1-21. doi:http://dx.doi.org/10.1186/s40537-016-0039-2

[19] Porter MF (1980) An algorithm for suffix stripping. Program 14 (3):130-137. Doi: doi: 10.1108/eb046814

[20] de Marneffe M-C, MacCartney B, Manning CD (2006) Generating typed dependency parsers from phrase structure parses Paper presented at the fifth international conference on language resources and evaluation, GENOA , ITALY 22-28 May 2006

[21] Blei DM, Lafferty JD (2005) Correlated Topic Models. Paper presented at the Proceedings of the 19th Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, 5-8 Dec. 2005

[22] Andrzejewski D, Zhu X, Craven M (2009) Incorporating domain knowledge into topic modeling via Dirichlet Forest priors. Paper presented at the Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, Quebec, Canada, 14-18 Jun. 2009

[23] Pedersen T, Patwardhan S, Michelizzi J (2004) WordNet::Similarity: measuring the relatedness of concepts. Paper presented at the Demonstration Papers at Human Language Technology conference/North American chapter of the Association for Computational Linguistics (HLT-NAACL), Boston, Massachusetts, USA, 2-7 May 2004

Authors

Ronald Brisebois is currently a PhD student at the École de Technologie Supérieure (ETS) – Université du Québec (Montréal, Canada). He received a B. Science in Physics at University of Montreal in 1983, a BA in Computer Science at University of Quebec in 1985 and his MBA at Hautes Études Commerciales - HEC (Business School) in 1989. From 1989 to 1995, Ronald Brisebois was a professor of Software Engineering at the University of Sherbrooke. His PhD research focus on semantic web, artificial intelligence, autonomous software architecture, new generation software designing, enriched metadata modeling and software engineering. Renowned entrepreneur in the field of information technology, Ronald Brisebois has held management positions in various top-level firms (Caisses populaires Desjardins). In 1991, he was a professor at the University of Sherbrooke; in 1992, he founded his first company. Cognicase Inc. quickly became one of the largest players in the information technology field in Canada. In 2003, Ronald created Isacsoft/MondoIn, one of the leading providers of integrated solutions for public libraries, academic institutions, specialized and consortia systems worldwide.

Dr. Abran holds a Ph.D. in Electrical and Computer Engineering (1994) from École Polytechnique de Montréal (Canada) and master degrees in Management Sciences (1974) and Electrical Engineering (1975) from University of Ottawa (Canada).He is a professor at the École de Technologie Supérieure (ETS) – Université du Québec (Montréal, Canada). He has over 20 years of experience in teaching in a university environment as well as 20 years of industry experience in information systems development and software engineering management. His research interests include software productivity and estimation models, software engineering foundations, software quality, software functional size measurement, software risk management and software maintenance management. He has published over 400 peer-reviewed papers. He is the author of the books 'Software Project Estimation', 'Software Metrics and Software Metrology' and a co--author of the book 'Software Maintenance Management' (Wiley Interscience Ed. & IEEE-CS Press). Dr. Abran is also the 2004 co-executive editor of the Guide to the Software Engineering Body of Knowledge – SWEBOK (see ISO 19759 and www.swebok.org) and he is the chairman of the Common Software Measurement International Consortium (COSMIC) – http://cosmic-sizing.org/. A number of Dr. Abran research works have influenced international standards in software engineering (i.e., ISO 19761, ISO 19759, ISO 14143-3, etc.)

Dr. Apollinaire Nadembega is currently a guest member of the Network Research Laboratory (NRL) of the University of Montreal. He received his B. E degree in Information Engineering from Computer Science High School, Bobo-Dioulasso, Burkina faso in 2003, his Master's degree in computer science from the Arts and Business Institute, Ouagadougou, Burkina faso in 2007 and his Ph.D. degree in mobile networks from the University of Montreal, Montreal, QC, Canada in 2014. The primary focus of his Ph.D. thesis is to propose a mobility model and bandwidth reservation scheme that supports quality-of-service management for wireless cellular networks. Dr. Nadembega's research interests lie in the field of artificial intelligence, machine learning, networking modelling, semantic web, metadata management system, software architecture, mobile multimedia streaming, call admission control, bandwidth management and mobile cloud computing. From 2004 to 2008, he was a programming engineer with Burkina Faso public administration staff management office.

Philippe started with a three-year training as a computer expert at the institute Leonardo da Vinci in Italy. Then, he joined the University of Parma, where he obtained his Bachelor in Computer Engineering with honors. He was then admitted at Polytechnic of Milan, one of the most prestigious engineering school (24th for Engineering in the world) for a master degree in computer engineering. After his first year, he won a scholarship for a double degree exchange program with the Polytechnic School of Montreal to obtain a second master more focused towards research in Natural Language Processing. In the last two years, he worked as research scientist for Ecole Polytechnique de Montreal, Bibliomondo and Nuance communications.

Paper 5:

A Semantic Metadata Enrichment Software Ecosystem based on Machine Learning to Analyse Topic, Sentiment and Emotions

Ronald Brisebois, Alain Abran, Apollinaire Nadembega, Philippe N'techobo

http://recentscientific.com/sites/default/files/7380-A-2017.pdf

ISSN: 0976-3031

Available Online at http://www.recentscientific.com

CODEN: IJRSFP (USA)

International Journal of Recent Scientific Research
Vol. 8, Issue, 4, pp. 16698-16714, April, 2017

International Journal of Recent Scientific Research

DOI: 10.24327/IJRSR

Research Article

A SEMANTIC METADATA ENRICHMENT SOFTWARE ECOSYSTEMBASED ON MACHINE LEARNING TO ANALYZE TOPIC, SENTIMENT AND EMOTIONS

Ronald Brisebois[1]., Alain Abran[1]., Apollinaire Nadembega*[2] and Philippe N'techobo[3]

[1]École de technologie supérieure, University of Quebec, Montreal, Canada
[2]Network Research Lab., University of Montreal, Montreal, Canada
[3]École Polytechnique de Montréal, Montreal, Canada

DOI: http://dx.doi.org/10.24327/ijrsr.2017.0804.0200

ARTICLE INFO

Article History:

Received 06th January, 2017
Received in revised form 14th February, 2017
Accepted 23rd March, 2017
Published online 28th April, 2017

Key Words:

Emotion detection, natural language processing, semantic topic detection, semantic metadata enrichment, sentiment analysis, text and data mining.

ABSTRACT

In a previous paper, a semantic metadata enrichment software ecosystem(SMESE) based on a multi-platform metadata model and a hybrid machine learning model have been proposed. This work presents the SMESE V3 version enhanced with interest-based enrichments through text analysis approaches for sentiments/emotions detection and hidden topics discovery. SMESE V3 makes it possible to create and use a semantic master catalogue with enriched metadata that allows interest-based search and discovery.

This paper presents the design, implementation and evaluation of a the SMESE V3platform using metadata and data from the web, linked open data, harvesting and concordance rules, and bibliographic record authorities. The SMESE V3 platform includes three distinct engines that:

1. Identify and enrich sentiment/emotion metadata hidden within the text or multimedia structure using the proposed a new BM-Semantic Sentiment and Emotion Analysis algorithm.
2. Propose an hybrid machine learning model for metadata enrichment.
3. Generate semantic to pics by text, and multimedia content analysis using the proposed BM-Scalable Annotation-based Topic Detection algorithm.

The performance of SMESE V3is evaluated using a number of prototype simulations by comparing them to existing enriched metadata technique and classifications. The results show that the enhanced SMESE V3 and related algorithms allow greater performance for purposes of interest-based search.

INTRODUCTION

The rapid development of search and discovery engines, the sudden availability of millions of documents, and the millions upon millions of relationships to linked documents from a growing multitude of sources (e.g., online media, social media and published documents) all make it challenging for a user to find documents relevant to his or her interests or emotions.

The human brain has an inherent ability to detect topics, emotions, relationships or sentiments in written or spoken language. However, the internet, social media and repositories have expanded the number of sources, volume of information and number of relationships so fast that it has become difficult to process all this information[1]. The goal is to increase the find ability of entities matching user interest using external (outside documents) and internal (within documents) semantic metadata enrichment algorithms. While computer search engines struggle to understand the meaning of natural language, semantically enriching entities with meaningful metadata may improve those capabilities. Words themselves are often used inconsistently, having a wide variety of definitions and interpretations. Although there may beno relationship between individual words of a topic or sentiment/emotion, thesauri do express associative relationships between words, ontologies, entities and a multitude of relationships represented as triplets. Finding bibliographic references or semantic relationships in texts makes it possible to localize specific text segments using ontologies to enrich a set of semantic metadata related to topics or sentiments/emotions. The current methodology proposed by researchers for text analysis within the context of entitymeta data enrichment (EME) reduces each document in the corpus to

*Corresponding author: *Apollinaire Nadembega*
École de technologie supérieure, University of Quebec, Montreal, Canada

a vector of real numbers where each vector represents ratios of counts. Several EME approaches have been proposed, most of them making use of term frequency-inverse document frequency (tf-idf)[2, 3]. In the tf-idf scheme, a basic vocabulary of "words" or "terms" is chosen, then for each document in the corpus, a frequency count is calculated from the number of occurrences of each word [2, 3]. After suitable normalization, the frequency count is compared to an inverse document frequency count (e.g the inverse of the number of documents in the entire corpus where a given word occurs-generally on a log scale, and again suitably normalized). The end result is a term-by-document matrix X whose columns contain the tf-idf values for each of the documents in the corpus. Thus the tf-idf scheme reduces documents of arbitrary length to fixed-length lists of numbers. For non-textual content, tools are available to extract the text from multimedia entities. Bougiatiotis and Giannakopoulos[4]propose an approach that extracts topical representations of movies based on the mining of subtitles.

In the context of this work, we focus on two research axis of the EME research field: Semantic topic detection (STD) and sentiment/emotion analysis (SEA).

On the one hand, STD helps users to efficiently detect meaningful topics. It has attracted significant research in several communities in the last decade, including public opinion monitoring, decision support, emergency management and social media modeling [5, 6]. STD is based on large and noisy data collections such as social media, and addresses both scalability and accuracy challenges. Initial methods for STD relied on clustering documents based on a core group of keywords representing a specific topic, where, based on a ratio such as tf-idf, documents that contain these keywords are similar to each other [2, 3]. Next, variations of tf-idf were used to compute keyword-based feature values, and cosine similarity was used as a similarity (or distance) measure to cluster documents. The subsequent generation of STD approaches, including those based on latent Dirichl *et al* location (LDA), shifted analysis from directly clustering documents to clustering keywords. Some examples of these advances in STD are presented in [7].

However, social media collections differ along several criteria, including the size distribution of documents and the distribution of words. One challenge is to rapidly filter noisy and irrelevant documents, while at the same time accurately clustering a large collection. Bijalwan *et al*. [8], for example, experimented with machine learning approaches for text and document mining and concluded that k-nearest neighbors(KNN), for their data sets, showed the maximum accuracy as compared to naive Bayes and term-graph. The drawback for KNN is that time complexity (i.e., amount of time taken to run) is high but it demonstrates better accuracy than others.

On the other hand, the main objective of sentiment analysis (SA) is to establish the attitude of a given person with regard to sentences, paragraphs, chapters or documents [1, 3, 9-15]. Indeed, many websites offer reviews of items like books, cars, mobiles, movies etc., where products are described in some detail and evaluated as good/bad, preferred/not preferred; unfortunately, these evaluations are insufficient for users in order to help them to make decision. In addition, with the rapid spread of social media, it has become necessary to categorize these reviews in an automated way[3].

For this automatic classification, there are different methods to perform SA, such as keyword spotting, lexical affinity and statistical methods. However, the most commonly applied techniques to address the SA problem belong either to the category of text classification supervised machine learning (SML), which uses methods like naïve Bayes, maximum entropy or support vector machine (SVM), or to the category of text classification unsupervised machine learning (UML). Also, fuzzy sets appear to be well-equipped to model sentiment-related problems given their mathematical properties and ability to deal with vagueness and uncertainty-characteristics that are present in natural languages processing. Thus, a combination of techniques may be successful in addressing SA challenges by exploiting the best of each technique. In addition, the semantic web may be a good solution for searching relevant information from a huge repository of unstructured web data [9].

One current limitation in the area of SA research is its focus on sentiment classification while ignoring the detection of emotions. For example, document emotion analysis may help to determine an emotional barometer and give the reader a clear indication of excitement, fear, anxiety, irritability, depression, anger and other such emotions. For this reason, we focus on sentiment/emotion analysis (SEA) instead of SA.

A number of algorithms or approaches are used to perform text mining, including: latent Dirichl *et al* location (LDA) [7], tf-idf[2, 3], latent semantic analysis (LSA)[16],formal concept analysis (FCA)[17], latent tree model (LTM)[18], naïve Bayes (NB)[19], support vector machine method (SVM) [19], artificial neural network (ANN)[20] based on the associated document's features.

Our approach improves the accuracy of topic detection and sentiment/emotion discovery by semantically enriching the metadata from the linked open data and the bibliographic records existing in different formats .This paper presents the design, implementation and evaluation of anenhancedeco system, called semantic metadata enrichment ecosystem or SMESE V3. Notice that SMESE V3 is an extension of our previous work on SMESE [21].

More specifically, SMESE V3consists of engines implementing two rule-based algorithms to enrich metadata semantically:

1. BM-SATD: generation of semantic topics by text analysis, relationships and multimedia contents.
2. BM-SSEA: discovery of sentiments/emotions hidden within the text or linked to a multimedia structure through an AI computational approach.

Using simulation, the performance of SMESE V3was evaluated in terms of accuracy of topic detection and sentiment/emotion discovery. Existing approaches to enriching metadata (e.g., topic detection or sentiment/emotion discovery) were used for comparison. Simulation results showed that SMESE V3 outperforms existing approaches.

International Journal of Recent Scientific Research Vol. 8, Issue, 4, pp. 16698-16714, April, 2017

The remainder of the paper is organized as follows. Section 2 presents the related work. Section 3 describes SMESE V3 and its various algorithms while Section 4 presents the prototype of the SMESE V3 multiplatform architecture developed. Section 5presents the evaluation through a number of simulations. Section 6 presents a summary and some suggestions for future work.

RELATED WORK

Interest in entity metadata extraction was initially limited to those in the community who preferred to concentrate on manual design of on tologies as a measure of quality. Following the linked data bootstrapping provided by DB pedia, many changes ensued with a related need for substantial population of knowledge bases, schema induction from data, natural language access to structured data, and in general all applications that make for joint exploitation of structured and unstructured content. In practice, Graph-based methods, meanwhile, were incrementally entering the toolbox of semantic technologies at large.

Topic detection

In the last decade, semantic topic detection has attracted significant research in several communities, including information retrieval. Generally, a topic is represented as a set of descriptive and collocated keywords/terms. Initially, document clustering techniques were adopted to cluster content-similar documents and extract keywords from clustered document sets as the representation of topics (subjects). The predominant method for topic detection is the latent Dirichl *et al* location (LDA) [7], which assumes a generating process for the documents. LDA has been proven a powerful algorithm because of its ability to mine semantic information from text data. Terms having semantic relations with each other are collected as a topic. LDA is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities. In the context of text modeling, topic probabilities provide an explicit representation of a document.

The literature presents two groups of text-based topic detection approaches based on the size of the text: short text [17, 22-24]such as tweets or Facebook posts, and long text [4, 5, 7, 18, 25, 26] such as a book. For example, Dang *et al.*[22] proposed an early detection method for emerging topics based on dynamic Bayesian networks in micro-blogging networks. They analyzed the topic diffusion process and identified two main characteristics of emerging topics, namely attractiveness and key-node. Next, based on this identification, they selected features from the topology properties of topic diffusion, and built a DBN-based model using the conditional dependencies between features to identify the emerging keywords. But to do so, they had to create a term list of emerging keyword candidates by term frequency in a given time interval.

Cigarran *et al.* [17] proposed an approach based on formal concept analysis (FCA). Formal concepts are conceptual representations based on the relationships between tweet terms and the tweets that have given rise to them.

Cotelo *et al.* [23], when addressing the tweet categorization task, explored the idea of integrating two fundamental aspects of a tweet: the textual content itself, and its underlying structural information. This work focuses on long text topic detection.

Recently, considerable research has gone into developing topic detection approaches using a number of information extraction techniques (IET), such as lexicon, sliding window, boundary techniques, etc. Many of these techniques [5, 18, 25, 26]rely heavily on simple keyword extraction from text.

For example, Sayyadi and Raschid [5] proposed an approach for topic detection, based on keyword-based methods, called KeyGraph, that was inspired by the keyword co-occurrence graph and efficient graph analysis methods.

In other words, KeyGraph is based on the similarity of keyword extraction from text. We note two limitations to the approach, which requires improvement in two respects. Firstly, they failed to leverage the semantic information derived from topic model. Secondly, they measured co-occurrence relations from an isolated term-term perspective; that is, the measurement was limited to the term itself and the information context was overlooked, which can make it impossible to measure latent co-occurrence relations.

Salatino and Motta [26]suggested that it is possible to forecast the emergence of novel research topics even at an early stage and demonstrated that such an emergence can be anticipated by analyzing the dynamics of pre-existing topics.

Sentiment analysis (SA)

There are three main techniques for sentiment analysis (SA)[27]:keyword spotting. lexical affinity and statistical methods. The first two methods are well known while statistical methods have to be more explored further.

Statistical methods, such as Bayesian inference and support vector machines, are supervised approaches in which a labeled corpus is used for training a classification method which builds a classification model used for predicting the polarity of novel texts. By feeding a large training corpus of affectively annotated texts to a machine learning algorithm, it is possible for the system to not only learn the affective valence of related keywords (as in the keyword spotting approach), but also to take into account the valence of other arbitrary keywords (like lexical affinity), punctuation, and word co-occurrence frequencies. Sentiment analysis can be carried out at different levels of text granularity: document [19, 28-32], sentence[1, 3, 9, 33, 34], phrase [35], clause, and word [20, 36, 37]. Sentiment analysis may be at the sentence or phrase level (which has recently received quite a bit of research attention) or at the document level.

In [11], the authors presented a survey of over one hundred studies published in the last decade on the tasks, approaches, and applications of sentiment analysis. With a major part of available worldwide data being unstructured (such as text, speech, audio, and video), this poses important research challenges. In recent year numerous research efforts have led to automated SEA, an extension of the NLP area of research.

The first five dimensions represent tasks to be performed in the broad area of SEA. For the first three dimensions (subjectivity classification, sentiment classification and review usefulness

measurement), the authors note that the applied approaches are broadly classified into three categories: machine learning, lexicon based and hybrid approaches.

Since one of our research objectives was to extract sentiment and emotion metadata from documents, the rest of this section focuses on sentiment classification, lexicon creation, and opinion word and product aspect extraction. Sentiment classification is concerned with determining the polarity of a sentence; that is, whether a sentence is expressing positive, negative or neutral sentiment towards the subject. A lexicon is a vocabulary of sentiment words with respective sentiment polarity and strength value while opinion word and product aspect extraction is used to identify opinion on various parts of a product. For the purpose of this paper, we assume that a document expresses sentiments on a single content and is written by a single author.

Cho *et al.* [30]proposed a method to improve the positive vs. negative classification performance of product reviews by merging, removing, and switching the entry words of the multiple sentiment dictionaries. They merge and revise the entry words of the multiple sentiment lexicons using labeled product reviews. Specifically, they selectively remove the sentiment words from the existing lexicon to prevent erroneous matching of the sentiment words during lexicon-based sentiment classification. Next, they selectively switch the polarity of the sentiment words to adjust the sentiment values to a specific domain. The remove and switch operations are performed using the target domain's labeled data (i.e. online product reviews) by comparing the positive and negative distribution of the labeled reviews with a positive and negative distribution of the sentiment words. They achieved 81.8% accuracy for book reviews. However, their contribution is limited to development of a novel method of removing and switching the content of the existing sentiment lexicons.

Moraes *et al.* [19] compared well-known machine learning approaches (SVM and NB) with an ANN-based method for document-level sentiment classification. Naive Bayes (NB) is a probabilistic learning method that assumes terms occur independently while the support vector machine method (SVM) seeks to maximize the distance to the closest training point from either class in order to achieve better generalization/classification performance on test data. The authors reported that, despite the low computational cost of the NB technique, it was not competitive in terms of classification accuracy when compared to SVM. According to the authors, many researchers have reported that SVM is perhaps the most accurate method for text classification. Artificial neural network (ANN) derives features from linear combinations of the input data and then models the output as a nonlinear function of these features. Experimental results showed that, for book datasets, SVM outperformed ANN when the number of terms exceeded 3,000. Although SVM required less training time, it needed more running time than ANN. For 3,000 terms, ANN required 15 sec training time (with negligible running time) while SVM training time was negligible (1.75 sec). In addition, their contribution was limited to performing comparisons between existing approaches. As in [19], Poria S. *et al.*[38] experimented with existing approaches and showed that SVM is a better approach for text-based emotion detection.

Emotion analysis

Emotions are also associated with mood, temperament, personality, outlook and motivation [27, 39, 40]. However, sentiments are differentiated from emotions by the duration in which they are experienced. The SWAT model was proposed to explore the connection between the evoked emotions of readers and news headlines by generating a word-emotion mapping dictionary. For each word w in the corpus, it assigns a weight for each emotion e, i.e., $P(e|w)$ is the averaged emotion score observed in each news headline H in which w appears. The emotion-term model is a variant of the NB classifier and was designed to model word-emotion associations. In this model, the probability of word wj conditioned on emotion ek is estimated based on the co-occurrence count between word wj and emotion ek for all documents. The emotion-topic model is combination of the emotion-term model and LDA.

A system for text-based emotion detection is proposed by Anusha and Sandhya [41]which uses a combination of machine learning and natural language processing techniques. They used the Stanford CoreNLP toolkit to create the dependency tree based on word relationships. Phrase selection is done using the rules on dependency relationships that gives priority to the semantic information for the classification of a sentence's emotion. Next, they used the Porter stemming algorithm for stemming, and stop words removal and tf-idf to build the feature vectors.

Cambria *et al.*[42]explored how the high generalization performance, low computational complexity, and fast learning speed of extreme learning machines can be exploited to perform analogical reasoning in a vector space model of affective common-sense knowledge. After performing TSVD on Affect Net, they used the Frobenius norm to derive a new matrix. For the emotion categorization model, they used the Duchenne smile and the Klaus Scherer model.

Conclusion

Some of our key findings from the related work on sentiment and emotion analysis are:

1. Traditional sentiment analysis methods mainly use terms and their frequency, part of speech, rule of opinions and sentiment shifters. Semantic information is ignored in term selection, and it is difficult to find complete rules.
2. Most of the recent contributions are limited to sentiment analysis elaborated in terms of positive or negative opinion and do not include the analysis of emotions.
3. Existing approaches do not take into account the human contribution to improve accuracy.
4. Existing approaches do not combine sentiment and emotion analysis.
5. Lexicon and ontology based approaches provide good accuracy for text-based sentiment and emotion analysis when applying SVM techniques. In our work, it is more important to identify the sentiment and emotion of a book taking into account all the books of the collection. For example, assuming that book A has 90% fear and 80% sadness while the emotion which has the best weight of book B is 40% fear; can it be said that fear is the emotion of book B as in book A?

International Journal of Recent Scientific Research Vol. 8, Issue, 4, pp. 16698-16714, April, 2017

6. Existing approaches do not take into account document collections. In terms of granularity, most of the existing approaches are sentence-based.
7. These approaches do not take into account the context around the sentence and in this way, it is possible to miss the real emotion.

As a general conclusion to the literature review on topic detection, sentiment and emotion analysis, 95% of the work focused on features of the documents (e.g., sentence length, capitalized words, document title, term frequency, and sentences position) to perform text mining and generally make use of existing algorithms or approaches (e.g., LDA, tf-idf, VSM, SVD, LSA, TextRank, PageRank, LexRank, FCA, LTM, SVM, NB and ANN) based on their features associated to documents.

Table I compares the most known text mining algorithms (e.g., AlchemyAPI, DBpedia, Wikimeta, open calais, Bitext, AIDA, TextRazor) with our algorithms proposed in SMESE V3by keyword extraction, classification, sentiment analysis, emotion analysis and concept extraction.

RULE-BASED SEMANTIC METADATA INTERNAL ENRICHMENT ENGINE

This section presents an overview and the details of the proposed a rule-based semantic metadata internal enrichment engine, a Machine Learning Engine (MLE), including two different algorithms (BM-SATD and BM-SSEA).

MLE is part of the SMESE V3platform architecture as shown in Fig. 1. The main goal of SMESE V3 is to enhance the SMESE platform through text analysis approaches for topics, sentiment/emotion and semantic relationships detection. SMESE V3 allows one to create a semantic master catalogue with enriched metadata that enables the search and discovery interest-based engines. To perform this task, the following tools are needed:

1. Topics are a controlled set of terms designed to describe the subject of a document. While topics do not necessarily include relationships between terms, we include relationships as triplets (Entity - Relationship - Entity).
2. A multilingual thesauri and ontology to provide hierarchical relationships as well as semantic relationships between topics.

Table I Summary of attribute comparison of existing and proposed algorithms

Existing algorithms	Keyword extraction	Classification	Sentiment analysis	Emotion analysis	Concept extraction
AlchemyAPI (http://www.alchemyapi.com/)	x	x	x	x	x
DBpedia Spotlight (https://github.com/dbpedia-spotlight)					x
Wikimeta (https://www.w3.org/2001/sw/wiki/Wikimeta)					x
Yahoo! Content Analysis API (out of date) (https://developer.yahoo.com/contentanalysis/)		x			x
Open Calais (http://www.opencalais.com/)	x	x			x
Tone Analyzer (https://tone-analyzer-demo.mybluemix.net/)			x	x	
Zemanta (http://www.zemanta.com/)					x
Receptiviti (http://www.receptiviti.ai/)			x	x	
Apache Stanbol (https://stanbol.apache.org/)					x
Bitext (https://www.bitext.com/)			x		x
Mood patrol (https://market.mashape.com/soulhackerslabs/moodpatrol-emotion-detection-from-text)				x	
Aylien (http://aylien.com/)	x	x	x		
AIDA (http://senseable.mit.edu/aida/)					x
Wikifier (http://wikifier.org/)					x
TextRazor (https://www.textrazor.com/)					x
Synesketch (http://krcadinac.com/synesketch/)				x	
Toneapi (http://toneapi.com/)			x	x	
SMESE V3	x	x	x	x	x

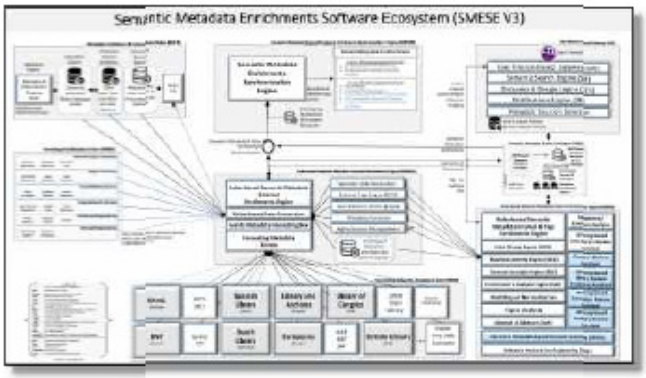


Fig 1 SMESE V3-Semantic Metadata Enrichment Software Ecosystem

3. Anontology to provide a representation of knowledge with rich semantic relationships between topics. By breaking content into pieces of data, and curating semantic relationships to external contents, metadata enrichments are created dynamically.

In Fig. 1, the V3 improvements to the SMESE platform from this work and its implementation are presented in blue.

The following sub-sections present the terminology and assumptions, the necessary pre-processing and details of the two algorithms proposed and implemented.

Terminology and assumptions

In this section the following terms are defined:

1. A word or term is the basic unit of discrete data, defined to be an item from a vocabulary indexed by {1, …,V}. Terms are presented using unit-basis vectors. Thus, the ith term in the vocabulary is represented by an I-vector w such that $w^i = 1$ and $w^j = 0$ for $i \neq j$.
2. A lineis a sequence of N terms denoted by l.
3. A document is a sequence of N lines denoted by D = $(w_1,w_2; \ldots,w_N)$, where w_i is the ith term in the sequence coming from the lines. D is represented by its lines as $D = (l_1, \ldots l_i, \ldots, l_K)$.
4. A corpus is a collection of M documents denoted by $C=\{D_1, D_2, \ldots, D_M\}$.
5. An emotion word is a word with strong emotional tendency or a probabilistic distribution.

To implement the BM-SATD and BM-SSE Aalgorithms, machine learning models have been used to perform metadata enrichments (see Fig.2):

3. A Machine Learning Engine allows to use a combination of supervised and unsupervised and allows to generate a predictive model
4. A feedback processing allows to the Machine Learning Engine to learn.
5. New texts or documents who are converted into Metadata vectors use the predictive model generated in 3.

Document pre-processing

The objective of the pre-processing is to filter noise and adjust the data format to be suitable for the analysis phases. It consists of stemming, phase extraction, part-of-speech filtering and removal of stop-words. The corpus of documents crawled from specific databases or the internet consists of many documents. The documents are pre-processed into a basket dataset C, called the document collection. C consists of lines representing the sentences of the documents. Each line consists of terms, i.e. words or phrases. More specifically, a pre-processing including tokenization, lower casing and stemming of all the terms using the Porter stemmer[43] is performed.

Scalable annotation-based topic detection: BM-SATD

The aim of BM-SATD is to build a classifier that can learn from already annotated contents (e.g., documents and books) and infer the topics of new books. Traditional approaches are typically based on various topic models, such as latent Dirichl *et al* location (LDA) where authors cluster terms into a topic by mining semantic relations between terms. However, co-occurrence relations across the document are commonly neglected, which leads to detection of incomplete information.

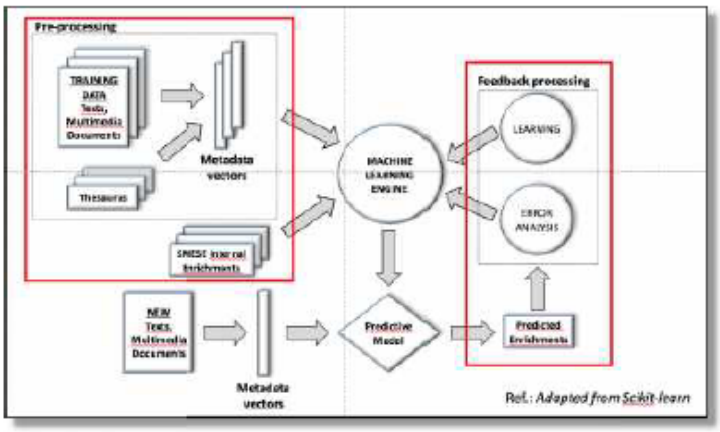


Fig. 2 Supervised Learning applied to Metadata Enrichments

1. There is a pre-processing using Training Data.
2. One or multiple thesaurus are available. A thesaurus contains a list of words with synonyms and related concepts. This approach uses synonyms or glosses of lexical resources in order to determine the emotion or polarity of words, sentences and documents.

Furthermore, the inability to discover latent co-occurrence relations via the context or other bridge terms prevents important but rare topics from being detected. BM-SATD combines semantic relations between terms and co-occurrence relations across the document making use of document annotation. In addition, BM-SATD includes:

International Journal of Recent Scientific Research Vol. 8, Issue, 4, pp. 16698-16714, April, 2017

1. A probabilistic topic detection approach, called semantic topic model (BM-SemTopic).
2. A clustering approach that is an extension of KeyGraph, called semantic graph (BM-SemGraph).

BM-SATD is a hybrid relation analysis and machine learning approach that integrates semantic relations, semantic annotations and co-occurrence relations for topic detection. More specifically, BM-SATD fuses multiple relations into a term graph and detects topics from the graph using a graph analytical method. It can detect topics not only more effectively by combining mutually complementary relations, but it can also mine important rare topics by leveraging latent co-occurrence relations. The following sub-sections present the details of the five phases of the BM-SATD model.

Relevant and less similar documents selection

A filtering process is performed to avoid using a large corpus of documents that are similar or not relevant. It is not necessary to compare a new document of a collection with two other documents of the collection that are similar in order to know whether this new document is similar to each of the other documents. This strategy merely increases computation time. Here, only documents that are already annotated by topic are considered.

Not annotated documents semantic term graph generation

The semantic term graph is a basis for detecting topics automatically. The BM-SemGraph has one node for each term in the vocabulary of the document. Edges in a BM-Sem Graph represent the co-occurrence of the corresponding keywords and are weighted by the count of the co-occurrences. Note that, in contrast to existing graph-based approaches, the co-occurrence between A and B is different from the co-occurrence between B and A. This difference allows one to retain the semantic sense of co-occurrence terms.

Step 1: Co-occurrence clusters generation

For the co-occurrence graph, the assumption is that terms that have a close relation to each other may be linked by the co-occurrence link. The relation between two terms W_i and W_j is measured by their conditional probability. Let D be a document and $V_D = (w_1, w_2; \ldots, w_N)$ be the terms of D and L_D be the number of lines of D.

The conditional probability $p(W_i, W_j^{\varepsilon})$ of W_i, W_j^{ε} is computed using equation (1)where:

1. ε denotes the minimum distance between W_i and W_j
2. The distance between two terms is the number of terms that appear between them for a given line
3. ε is a parameter determined by experimentation.

$$p\left(W_i, W_j^{\varepsilon}\right) = \sum_{l=1}^{L_D} \frac{N^{line\ l}(W_i, W_j^{\varepsilon})}{\frac{N(line\ l)}{\varepsilon}} \quad (1)$$

where $N^{line\ l}\left(W_i, W_j^{\varepsilon}\right)$ denotes the number of times that W_i and W_j co-occur with a minimum distance ε and where W_i appears before W_j, and *N(line l)* denotes the number of terms of the line l.

To formally define a relation between two terms W_i and W_j, their frequent co-occurrence measured by the conditional probability $p(W_i, W_j^{\varepsilon})$, needs to exceed the co-occurrence threshold. The co-occurrence threshold is also determined by experimentation. Note that frequent co-occurrence is oriented. This allows one to retain the semantic orientation of the links between terms. Next, the oriented links are transformed into simple links without losing the semantic context.

Step 2: Cluster optimization

To improve quality, clusters should be pruned, such as by removing weak links or partitioning sparse cluster into cohesive sub-clusters. Clusters are pruned according to their connectedness. The link *e* is pruned when no path connects the two ends of *e* after it is pruned. The link between the black node and the green node should be pruned. Secondly, cliques are identified. Let C be the clique and W_i and W_j be the nodes of C that are linked to another node. The weight between W_i and W_j is given by equation (2):

$$w\left(W_i, W_j\right) = \max_{\substack{W_k \in C \\ W_s \in C}} w\left(W_k, W_s\right) \quad (2)$$

Step 3: Key term extraction

To extract key terms, the relation between a term and a cluster is measured. It is assumed that the weight of a term in a given cluster may be used to determine the importance of this term for the cluster. Let R be the set of nodes of the cluster C where the node W_i is inside. The weight of W_i in the cluster C is given by equation (3):

$$f\left(W_i\right) = \sum_{W_j \in R} w\left(W_i, W_j\right) \quad (3)$$

To identify a term as a key term, a sort of terms is performed based on their weights regardless of the clusters that they are in. Next, the Num KeyTerm terms that have the largest weights are selected as Key Terms. NumKeyTerm is a parameter.

Step 4: Semantic topic generation

Semantic topic generation combines a correlated topic model (CTM) [44]and a domain knowledge model (DKM) [45], called BM semantic topic model (BM-SemTopic), to build the real semantic topic model. In LDA, a topic is a probability distribution over a vocabulary. It describes the relative frequency each word is used in a topic. Each document is regarded as a mixture of multiple topics and is characterized by a probability distribution over the topics. A limitation of LDA is its inability to model topic correlation. This stems from the use of the Dirichlet distribution to model the variability among topic proportions. In addition, standard LDA does not consider domain knowledge in topic modeling.

To overcome these limitations, BM-SemTopic combines two models:

1. A correlated topic model (CTM)[44] that makes use of a logistic normal distribution.
2. A domain knowledge model (DKM)[45] that makes use of the Dirichlet distribution.

BM-SemTopic uses a weighted sum of CTM and DKM to compute the probability distribution of term W_i on the topic *z*. The sum is defined by equation (4):

$$h(W_i|z) = \omega CTM(W_i|z) + (1-\omega)\, DKM(W_i|z) \quad (4)$$

where ω is used to give more influence to one model based on the term distribution of topics.

When the majority of terms are located in a few topics, this means the domain knowledge is important and ω must be small. BM-SemTopic develops the CTM where the topic proportions exhibit a correlation with the logistic normal distribution and incorporates the DKM. A key advantage of BM-SemTopicis that it explicitly models the dependence and independence structure among topics and words, which is conducive to the discovery of meaningful topics and topic relations.

CTM is based on a logistic normal distribution. The logistic normal is a distribution on the simplex that allows for a general pattern of variability between the components by transforming a multivariate normal random variable. This process is identical to the generative process of LDA except that the topic proportions are drawn from a logistic normal distribution rather than a Dirichlet distribution.

DKM is an approach to incorporation of such domain knowledge into LDA. To express knowledge in anontology, BM-SemTopicuses two primitives on word pairs: Links and Not-Links. BM-Sem Topic replaces the Dirichlet prior by the Dirichlet Forest prior in the LDA model. Then, BM-Sem Topic sorts the terms for every topic in descending order according to the probability distribution of the topic terms. Nextit picks up the high-probability terms as the feature terms. For each topic, the terms with probabilities higher than half of the maximum probability distribution are picked up.

Step 5 Semantic term graph extraction

To discover semantic relations between the semantic terms, the semantic aspect is included making use of Word Net::Similarity [46]. Based on the structure and content of the lexical database Word Net, Word Net::Similarity implements six measures of similarity and three measures of relatedness. Measures of similarity use information found in a hierarchy of concepts that quantify how much concept A is like concept B.

When the semantic terms are identified, the semantic value of each topic's candidates is computed. The semantic value of each term W_i, is given by equation (5):

$$SEM(W_i|z) = TP - ITP(W_k|z) = h(W_i|z) * \log\left(\frac{|Z|}{\sum_{t \in z} h(W_i|t)}\right) \quad (5)$$

where Z denotes the set of semantic topics. TP-ITP is inspired by the tf-idf formula, where TP is term probability and ITP inverse topic probability.

Semantic links between semantic terms for the term graph are constructed using the vector measure, one of the measures of relatedness of Word Net::Similarity [46].The vector measure creates a co-occurrence matrix for each word used in Word Net glosses from a given corpus, and then represents each gloss/concept with a vector that is the average of these co-occurrence vectors.

Let W_i and W_j be semantic terms of the synsets A and B, respectively. Let $\vec{A} = (a_1, \ldots, a_q)$ and $\vec{B} = (b_1, \ldots, b_q)$ be the co-occurrence vectors of A and B, respectively. Let V_z be the set of semantic terms of the semantic topic Z. The weight of the link between W_i and $W_j$is computed by equation (6):

$$Dis(W_i, W_j|z) = \frac{SEM(W_i|z) + SEM(W_j|z)}{\frac{\sum_{W_k \in V_Z} SEM(W_k|z)}{n}} \times \sqrt{\sum_{l=1} (a_l - b_l)^2} \quad (6)$$

To discover a semantic relation between two terms, the semantic distance is computed. The semantic distance between two terms is the shortest path between the terms using equation (7):

$$SEMDis(W_i, W_j|z) = \min_{pa \in P}\left[\sum_{W_k \in pa} Dis(W_i, W_k|z)\right] \quad (7)$$

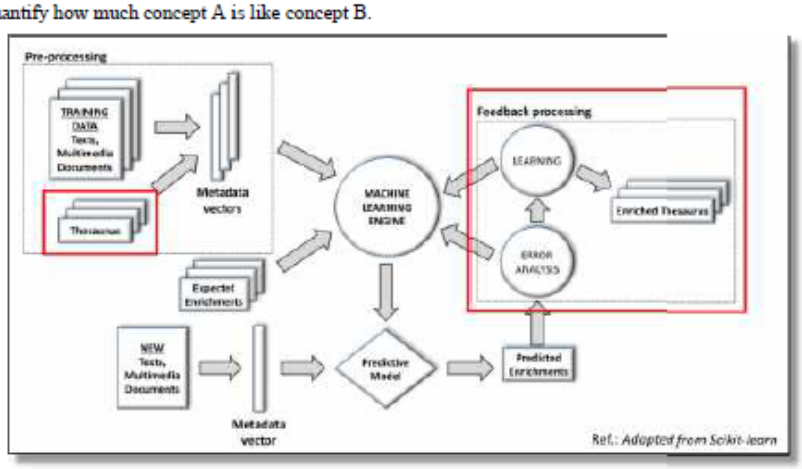


Fig. 3 Supervised Learning applied to Metadata Enrichments

International Journal of Recent Scientific Research Vol. 8, Issue, 4, pp. 16698-16714, April, 2017

where *pa*, W_k, and *P* denote a path between W_i and W_j in the thesaurus, a term on a path *pa* and the set of path *spa* between W_i and W_j, respectively. See Fig. 3, in the pre-processing phase, we can notice the usage of thesaurus. At the end of the machine learning process, an enriched thesaurus is generated to be part of the input of the machine learning process.

To formally define a semantic relation between two terms W_i and W_j, the semantic distance $SEMDis(W_i, W_j \mid z)$must not exceed the semantic threshold. The semantic threshold is determined by experimentation.

The last process to generate the semantic term graph BM-SemGraph is amerging of the term graph and the semantic graph. The term graph and semantic graph are merged by coupling the co-occurrence relation and the semantic relation. New terms are added as semantic terms and new links are added as semantic links if they do not appear in the term graph. For each link between two nodes W_j and W_k of the merged graph, the weight, called the BM Weight (BMW), for a given topic $t_i$is computed using equation (8):

$$BMW(W_j, W_k \mid t_i) = \frac{\lambda}{SEMDis(W_j, W_k \mid t_i)} + (1-\lambda) \times w(W_i, W_j) \quad (8)$$

where λdetermined by experimentation.

Topic detection

Topics that may be associated with a new document are detected based on the BM-SemGraph. Note that the BM-SemGraph is obtained using a collection of documents. In this case, the likelihood of detecting topics among a collection of documents is high and must be computed. To accomplish this, the feature vector of each topic based on the clusters of BM-SemGraphis computed. The feature vector of a topic is calculated using the BMRank of each topic term. Let A be the set of nodes of BM-SemGraph directly linked to term W_j in the topic t_i.The score forterm $W_j$is given by equation (9):

$$BMRank(W_j \mid t_i) = \frac{\sum_{W_k \in A} BMW(W_j, W_k \mid t_i)}{|A|} \quad (9)$$

The term with the largest BMRank is called the main term of the topic;the other terms are secondary terms. The same processes are used to obtain the BM-SemGraph of an individual document *d* and the feature vectors of topics t_j^d. Next, the similarity between each topic t_i and the topics t_j^d of document *d* is computed in order to detect document topics.

Training

The training process establishes a terms graph based on the relevant and less similar documents for a given topic t_i. To form the terms graph for a given topic, the pre-processing of its relevant and less similar documents is first carried out, a set of linesis obtained where each line is a list of terms, and the co-occurrence of these terms is then computed.

Topics refining

The architecture overview of the topic refining process phase in BM-SATD is presented in Fig. 4, this process refines the detected topics making use of relevant documents already annotated by humans based on existing or known topics. Following this process, three lists of topics are obtained: a list of new topics, a list of similar existing topics and a list of not similar existing topics.

The list of existing topics that match new document detected topics is identified based on the new document detected topics and annotated documents by topic (existing topics). Then, the clusters of terms by to pic are identified based on the collection of relevant and less similar documents. Note that each topic is a cluster of terms graph. Therefore, in this case, a graph matching technique is a good candidate to perform topic similarity detection.

Next, using our graph matching technique, the clusters of terms by topics of relevant and less similar collection of annotated documents which match with CTGare identified, for each cluster of terms graph by topic (CTG) of the new document. The matching score between two clusters is then computed. Let:

1. H be the new document terms graph and G be the terms graph obtained by a training process applied on the collection of relevant and less similar documents annotated by topics.
2. C_j^d be a cluster of H associated to topic t_j^d and C_i be a cluster of G associated with topic t_i.
3. W_i and W_j be two terms of cluster C_j^d; the link matching function $g(W_i W_j)$ between W_i and W_j is defined by equation (10):

$$g: C_j^d \times C_j^d \rightarrow IR$$

$$g(W_i W_j) = \begin{cases} MinHopCluster0fti(W_i W_j) & if\,path\,between\,W_i W_j \\ 1+MaxHopCluster0fti & if\,not\,path\,between\,W_i W_j \end{cases} \quad (10)$$

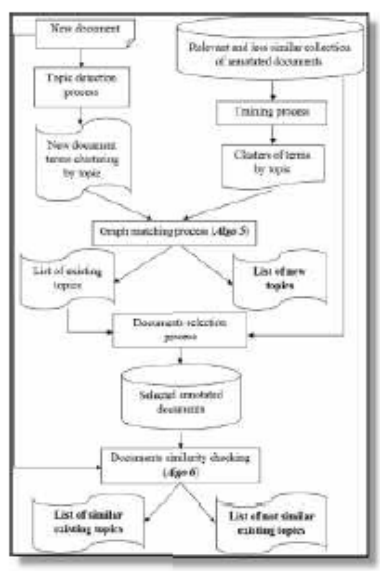


Fig 4 BM-SATD Topic refining process phase-Architecture overview

For a direct link $W_i^c W_j^c$ (only one hop between W_i and W_j) of cluster C_j^d, the process checks whether there is a path between W_i and W_j in the cluster C_i, regardless of the number of hops. Using the link matching function, the matching score between two clusters C_i^d and $C_j$is given by equation (11):

$$o : H \times G \rightarrow 0, 1$$

$$o\ C_j^d, C_i = \frac{C_j^d}{\sum_{W_i, W_j \in C_j^d} g(W_i^c W_j^c)} \quad (11)$$

where C_j^d is the number of links inclusters C_i^d.

Semantic sentiment and emotion analysis: BM-SSEA

The BM-SSEA goal is to classify the corpus of documents taking emotion into consideration, and to determine which sentiment it more likely belongs to. A document can be a distribution of emotion $p\ e|d\ \ e \in E$ and a distribution of sentiment $p\ s|d\ \ s \in S$.BM-SSEA is a hybrid approach that combines a keyword-based approach and a rule-based approach.BM-SSEA is applied at the basic word level and requires an emotional keyword dictionary that has keywords (emotion words) with corresponding emotion labels. To refine the detection, BM-SSEA develops various rules to identify emotion. Rules are defined using an affective lexicon that contains a list of lexemes annotated with their affect.

The emotional keyword dictionary and the affective lexicon are implemented in a thesaurus. BM-SSEA is a knowledge-based approach that uses an AI computational technique. The purpose of BM-SSEA is to identify positive and negative opinions and emotions.

For affective text evaluation, BM-SSEA uses the SS-Tagger (a part-of-speech tagger)[47] and the Stanford parser[48]. The Stanford parser was selected because it is more tolerant of constructions that are not grammatically correct. This is useful for short sentences such as titles. BM-SSEA also uses several lexical resources that create the BM-SSEA knowledge base located in the thesaurus.The lexical resources used are: WordNet, WordNet-Affect, SentiWordNet and NRC emotion lexicon. WordNet is a semantic lexicon where words are grouped into sets of synonyms, called synsets. WordNet-Affect is a hierarchy of affective domain labels that can further annotate the synsets representing affective concepts.

The NRC emotion lexicon is a thesaurus that associates for a word, the value one or zero for each emotion. This association is made of binary vectors. The disadvantage of this thesaurus is that since the values are binary, all words belonging to an emotion have the same weight for that emotion. To address this problem, the NRC emotion lexicon thesaurus was combined with the Word Net, WordNet-Affect and SentiWord Netthesaurus. This associates a feelings score with each word-POS. Where POS_1 are grammatical categories used to classify words in dimensions such as adjectives or verbs. Senti Word Net associates with each couple a valence score that can be either negative or positive with respect to the sense of the word in question. The word death, for example, is likely to have a negative score. BM-SSEA also relies on shifter valences.

For example, take the phrase "I am happy" witha score of 1 for the joy emotion. For the phrase "I am very happy", 'very' is a valence intensifier that will change the joy emotion score to 2. In the case, "I am **not** happy" the modifier 'not' will change the emotion joy to the contrary emotion sadness.

The main component of BM-SSEA is the thesaurus, called BMemotion word model (BMEmoWordMod). BMEmoWord Mod is an emotion-topic model that provides the emotional score of each keyword by taking the topic into account.

BMEmoWordMod introduces an additional layer (i.e., latent topic) into the emotion-term model such as SentiWordNet.BM-SSEA is composed of three phases: BMEmoWordMod generation process phase, sentiment and emotion discovery process phase and third sentiment and emotion refining process phase. The following sub-sections describe the three phases of the BM-SSEA model used to discover sentiment and emotion.

BMEmoWordMod generation process phase

A training set from the original corpusis created. The most relevant and discriminative documents are selected automatically. In the second step, each word is tagged with a POS and the combination of word and POS used as the essential feature. Finally, BMEmoWordMod is generated using the extracted features, which can then be used to discovery the sentiments and emotions of new documents. Many steps have to be completed: (1) Training set selection, (2) Intermediate lexicon generation and (3) Sentiment and emotion lexicon generation.

Sentiment and emotion discovery

This phase identifies the sentiments and emotions that are likely associated with a given new document by using the sentiment and emotion semantic lexicon BMEmo WordMod generated in the previous section. After preprocessing, the term vector of the new document is defined using TF-IDF.

Let ND be the new document and $W_{ND} = \{W_1, \ldots, W_z\}$ the set of distinct terms occuring in the corpus of documents. To obtain the z-dimensional term vector that represents each document in thecorpus, the tf-idf of each term of W_z is computed. The result of this computation establishes the term vector $\bar{t}_{ND} = (\text{tfidf}\ W_1, ND\ , \ldots, \text{tfidf}\ W_z, ND\ )$.

Using vector $\bar{t}_{ND}$, $T_{ND}=\{t_9, \ldots, t_q\}$ obtained using BM-SATD and BMEmoWordMod, the sentiment and emotion vector of new document

$E_{f_j,ND} =$
$(E\ f_j, ND, e_1\ , \ldots, E\ f_j, ND, e_E\ , E\ f_j, ND, s_P\ , E\ f_j, ND, s_N\ )$ is given by equation (12):

$$E\ f_j, ND, e_i = \frac{\text{tfidf}\ W_j, ND}{\sum_{i=1}^{z} \text{tfidf}\ W_i, ND} \times \sum_{t_k \in T_{ND}} \text{BMEmoWord}(f_j, e_i, t_k) \quad (12)$$

where $\text{BMEmoWord}(f_j, e_i, t_k)$ denotes the emotion probability of emotion e_i for the feature word f_j giving the topic t_k.$\text{BMEmoWord}(f_j, e_i, t_k)$ is selected in BMEmoWordMod.

International Journal of Recent Scientific Research Vol. 8, Issue, 4, pp. 16698-16714, April, 2017

The weight of emotion e_i for document ND is computed with equation (13):

$$W_E\ ND, e_i = \sum_{W_j \in W_{ND}} E\ f_j, ND, e_i\ ) \quad (13)$$

Equation (29) yields the emotional vector of new document ND

$$V_{ND} = W_E\ ND, e_1, \ldots, W_E\ ND, e_1, \ldots, W_E\ ND, e_E, W_E\ ND, S_P, W_E\ ND, S_N\ )).$$

Next, the new document ND emotion and sentiment is inferred using a fuzzy logic approach and the emotional vector $\overline{V}_{ND}$. The weight of emotion is transformed into five linguistic variables: very low, low, medium, high, and very high. Then, using these variables as input to the fuzzy inference system one obtains the final emotion for the new document.

Sentiment and emotion refining

The refining process validates discovered sentiment and emotion after the document analysis. Similarity is computed between new documents and documents in the corpus rated over E emotions. First, the term vectors of each document are defined using the tf-itf of each term, tf-itf is then computed using equation (1).Note that the terms extracted from the corpus of documents rated over E emotions are those employed by users. To measure the similarity between two documents, the cosine similarity of their representative vectors is computed. Two documents d1 and d2 are similar when the similarity $SimCos\ \overline{t}_{d1}, \overline{t}_{d2}$ of these two documents is less than the similarity threshold β.

EVALUATION USING SIMULATIONS

This section presents an evaluation of BM-SATD and BM-SSEA performance using simulations. To perform these simulations, an experimental environment was developed to provide a simulator to prototype the different algorithms of SMESE V3.

Dataset and parameters

To evaluate BM-SATD and BM-SSEA, real datasets from different projects that have digital and physical library catalogues were used. These datasets, consisting of 25,000 documents with a vocabulary of 375,000 words, were selected using average TF-IDF. The documents covered 20 topics and 8 emotions. The number of documents per topic or emotion was approximately equal. The average number of topics per document was 7 while the average rating emotion number per document was 4.15,000 documents of the dataset were used for the training phase and the remaining 10,000 other documents used for the test. Note that the 10,000 documents used for the tests were those that had more annotated topics or a higher rating over emotions.

To measure the performance of topic detection (sentiment and emotion discovery, respectively) approaches, comparison of detected topics (the discovered sentiment and emotion, respectively) with annotation topics of librarian experts (user ratings) were carried out. Table II presents the values of the parameters used in the simulations. The server characteristics for the simulations were: Dell Inc. Power Edge R630 with96 Ghz (4 x Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz, 10 core and 20 threads per CPU) and 256 GB memory running VMWare ESXi 6.0.

Table II Simulation parameters

Parameter	Value	Parameter	Value
ε	3	α	100
NumKeyTerm	8	co-occurrence threshold	0.75
ϖ	0.5	semantic threshold	1
β	0.7	term cluster matching threshold	0.45
λ	0.6		

Performance criteria

The performance of BM-SATD and BM-SSEA performance was measured in terms of running time[18] and accuracy[25][5]. Note that in the library domain, the most important criteria was precision while resource consumption was important for the software providers.

The running time, denoted by Rt, was computed as follows:

$$Rt = Et - Bt$$

where *Et* and denotes the time when processing is completed and *Bt* the time when it started.

To compute the accuracy, let $T_{annotated}$ and $T_{detected}$ be the set of annotated topic and the set of detected topics by BM-SATD for a given document d. The accuracy of topics detection, denoted by A_d^t, was computed as follows:

$$A_d^t = \frac{2.\,|T_{annotated} \cap T_{detected}|}{|T_{annotated}| + |T_{detected}|}$$

The same formula was applied to compute the accuracy of the sentiment and emotion discovery measurement. E_{rating}(resp. $E_{discovered}$) that denotes the set of rating over emotion (resp. the set of discovered emotion by BM-SSEA) was used instead of $T_{annotated}$ (resp. $T_{detected}$).

Simulation results were averaged over multiple runs with different pseudorandom number generator seeds. The average accuracy, *Ave_acc*, of multiple runs was given by:

$$Ave\_acc = \frac{\sum_{x=1}^{I} \frac{\sum_{d \in TD} A_d^t}{|TD|}}{I}$$

where TD denotes the number of tests documents and I denotes the number of test iterations.

The average running time, *Ave_run_time*, was given by:

$$Ave\_run\_time = \frac{\sum_{x=1}^{I} Rt}{I}$$

Topic detection approaches performance evaluation

BM-SATD performance was evaluated in terms of running time and accuracy. The dataset and parameters mentioned above were applied.BM-SATD performance was compared to the approaches described in [25], [5], [7] and [18], referred to as LDA-IG (probabilistic and graph approach), KeyGraph (graph analytical approach), LDA (probabilistic approach)and HLTM (probabilistic and graph approach), respectively.LDA-IG, Key Graph, LDA and HLTM were selected because they are text-based and long text approaches.

Comparison approaches

Table III presents the characteristics of the comparison approaches for topic detection.

The average relative improvement (defined as [Aver._runtime of BM-SATD −Aver._runtime of LDA]) of LDA compared with BM-SATD was approximately1.25 sec per document.

Table III Topic detection approaches for comparison

Approach	Granularity	Description	Training phase	Refining	Semantic	Topic correlation	Domain knowledge
LDA-IG [25]	Document	Probabilistic and graph based	Yes	No	No	No	No
KeyGraph [5]	Document	Graph based	Yes	No	No	No	No
LDA [7]	Document	Probabilistic based	No	No	No	No	No
HLTM [18]	Document	Probabilistic and graph based	Yes	No	No	No	No
BM-SATD	Configu-rable as desired	Semantic, probabilistic and graph based	Yes	Yes	Yes	Yes	Yes

Our proposed approach BM-SATD is the only one that is really semantic and takes into account the correlated topic and domain knowledge. The parameters for the comparison approaches used where those which provided the best performance.

Results analysis

Fig. 5presents the average running time of the detection phase when the number of documents used for the tests were varied. Training times were excluded as this phase was performed only one time. However, the BM-SATD training phase required more time than the other approaches. This was justified by the fact that BM-SATD identifies the relevant and less similar documents used for training phase. Fortunately, the new generation of data center equipment offers sufficient resources to reduce significantly the training delay. Only the time required to detect new document topics was measured.

Fig. 5 also shows that the average running time increased with the number of test documents. Indeed, the bigger the number of test documents, the longer the time to perform detection and, ultimately, the higher the average running time.

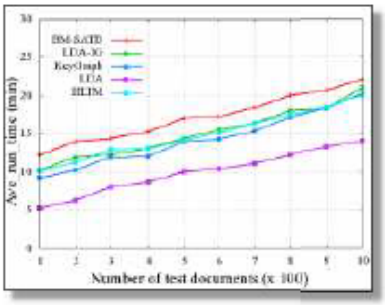


Fig. 5 Topic detection -Average running time versus number of documents for test phase

It was also observed that LDA outperforms the other approaches.LDA produced an average of 1.37 sec per document whereas BM-SATD produced an average of 2.62 secper document.

The short run times of LDA were due to the fact that LDA did not perform a graph treatment. Graph processing algorithms are very time consuming. Other approaches also outperformed BM-SATD on the running time criteria since BM-SATD performed topic refining in order to increase accuracy.

Fig. 6 shows the average accuracy when varying the number of detected topics. For the five approaches, the average accuracy decreased with the number of detected topics. The increase in the number of subjects to detect led to decreased accuracy. However, in terms of accuracy, BM-SATD outperformed the approaches used for comparison. BM-SATD produced an average accuracy of 79.50% per topic while LDA-IG, the best among the approaches used for comparison, produced an average of 61.01% per topic.

The average relative improvement in accuracy (defined as [Ave_acc of BM-SATD −Ave_acc of LDA-IG]) of BM-SATD compared to LDA-IG was 18.49% per topic. The performance of BM-SATD is explained as follows:

1. BM-SATD used the relevant documents for training phase.
2. BM-SATD refined its detection topic results by measuring new document similarity with relevant and less similar annotated documents.
3. BM-SATD combined correlated topic model and domain knowledge model instead of LDA.

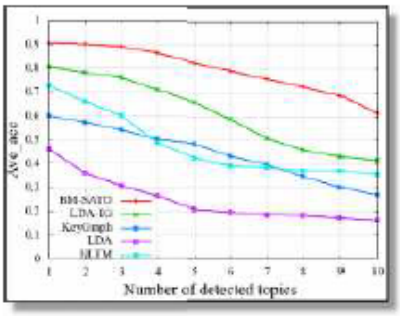


Fig. 6 Accuracy for number of detected topics for 5 comparison approaches

International Journal of Recent Scientific Research Vol. 8, Issue, 4, pp. 16698-16714, April, 2017

Fig. 6 also shows that BM-SATD produced an average accuracy of 90.32% for one detected topic and 61.27% for ten detected topics compared to 80.29% and 41.01% respectively for LDA-IG. The gap between BM-SATD accuracy and LDA-IG accuracy was 10.03% for one detected topic and 20.26% for ten detected topics. This meant that BM-SATD was by in large more accurate than LDA-IG in detecting several topics.

The Fig. 7presents the average accuracy when varying the number of training documents of the learning phase. LDA was not included in the scenario since not raining phase was performed. Fig. 7 shows that the average accuracy increased with the number of training documents. The larger the number of training documents, the better the knowledge about word distribution and co-occurrence and, ultimately, the higher the detection accuracy. However, the accuracy remained largely stable for very high numbers of training documents. When the number of documents of a collection was larger, the number of vocabulary words remained constant, and the term graph did not change. It also shows that HLTM was the approach whose detection accuracy was the first to reach stability at 10,000 training documents. HLTM builds a tree instead of a graph as the other approaches and its tree has less internal roots to identify topics. However, BM-SATD and LDA-IG outperformed HLTMin terms of accuracy.

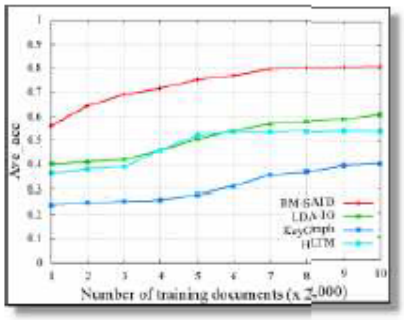


Fig. 7 Topic detection - accuracy for number of training documents

Fig. 7 also shows that BM-SATD outperformed LDA-IG on the accuracy criteria. For example, BM-SATD demonstrated an average accuracy of 73.49% per 2,000 training documents while LDA-IG produced an average accuracy of 50.86% per 2,000 training documents.The average relative improvement of BM-SATD compared to LDA-IG was 22.63% per 2,000 training documents. The better performance of BM-SATD followed from its use of a domain knowledge model. BM-SATD did not requirea large number of documents for the training phase. In conclusion, the 1.25 sec running time per document increase was a small price to pay for the larger average accuracy of topic detection (18.49%).

Sentiment and emotion analysis performance evaluation

BM-SSEA performance was also evaluated in terms of accuracy and running time. Simulations used the dataset and parameters mentioned previously. The performance of BM-SSEA was compared to the approaches described in [49] and [41], referred to as ETM-LDA and AP, respectively. ETM-LDA and AP were selected because they were document-based rather than phrase-based.

Comparison of approaches with BM-SSEA

Table IV shows the characteristics of the sentiment and emotion approaches used for comparison with BM-SSEA.

BM-SSEA was the only entirely semantic approach taking into account the rules for inferring emotion. In addition, BM-SSEA used a semantic lexicon. Several approaches used semantic lexicon, but these were limited to phrases rather than documents. The best performance approaches used were AP and ETM_LDA.

Results analysis

Fig. 8presents the average running time when varying the number of detected emotions. Training times were excluded because this phase was performed only once. The BM-SSEA training phase took more time than the other approaches due to lexicon aggregation and enrichment by users. The average running time increased with the number of test documents. This is normal, as the larger the number of test documents the longer the average running time to perform the sentiment and emotion discovery. Fig. 8 shows that ETM-LDA and AP outperformed BM-SSEA on the running time criteria. ETM-LDA required an average of 1.53 sec per document whereas BM-SSEA required an average of 1.74 sec per document. The average relative improvement of ETM-LDA compared with BM-SSEA was approximately 0.21 sec per document. The poorer performance of BM-SSEA resulted from refining sentiment and emotion to increase accuracy.

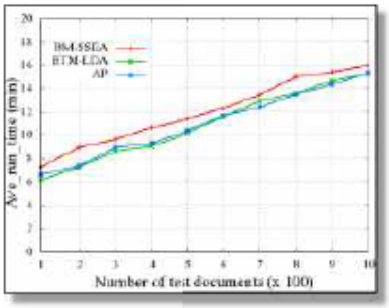


Fig. 8 Emotion discovery -Average running time versus number of documents for test phase

Table IV Sentiment and emotion approaches for comparison

Approach	Granularity	Approach	Training phase	Refi ning	The saurus	Topic modeling	Emotions number
AP [41]	Document	Learning based	Yes	No	5	No	8
ETM-LDA [49]	Document	Keyword based	Yes	No	6	Yes	8
BM-SSEA	Configurable as desired	Keywordand rule based	Yes	Yes	1,2,3, and 4	Yes	8

1-WordNet; 2-WordNet-Affect; 3-SentiWordNet; 4-NRC Emotion Lexicon; 5- Stanford Core NLP; 6-Gibbs sampling.

Fig. 9 presents the average accuracy when varying the number of discovered emotions. Positive and negative sentiments were not considered in the accuracy measurement. Fig. 9also shows that the average accuracy decreased with the number of discovered emotions. However, BM-SSEA outperformed the other two approaches used for comparisons. BM-SSEA demonstrated an average accuracy of 93.30% per emotion while ETM-LDA, the best of the other two approaches used for comparison, produced68.65% accuracy per emotion. The average relative improvement in accuracy of BM-SSEA compared to ETM-LDA was 24.65% per emotion. In conclusion, the 0.21 sec running time per document increase was, again, a small price to pay for the larger average accuracy of emotion discovery (24.65%).

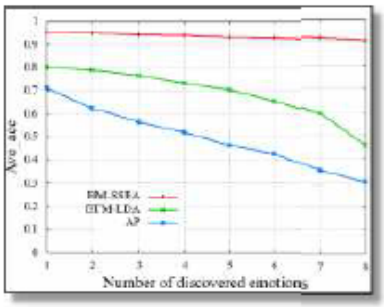


Fig. 9 Average detection accuracy for the number of discovered emotions

SUMMARY AND FUTURE WORK

In this paper, the goal was to increase the find ability (search, discover) of entities based on user interest using external and internal semantic metadata enrichment algorithms. As computers struggle to understand the meaning of natural language, enriching entities semantically with meaningful metadata can improve search engine capability. Words themselves have a wide variety of definitions and interpretations and are often utilized inconsistently. While topics and sentiments/emotions may have no relationship to individual words, thesauri express associative relationships between words, on tologies, entities and a multitude of relationships represented as triplets.

This paper has presented an enhanced V3 implementation of SMESE using metadata and data from the linked open data, structured data, metadata initiatives, concordance rules and authority metadata to create a master catalogue. It offers a foundation for an entire interest-based digital library of semantic mining activities, such as search, discovery and interest-based notifications. Finding bibliographic references or semantic relationships in texts makes it possible to localize specific text segments using on tologies to enrich a set of semantic metadata related to topic or sentiment/emotion.

To help users find interest-based contents, this paper has proposed an enhanced verions of the SMESE platform through text analysis approaches for sentiments/emotions detection. SMESE V3 can be used (or: makes it possible) to create and use a semantic master catalogue with enriched metadata that enables search and discovery interest-based engines. This paper has presented the design, implementation and evaluation of a SMESE V3 platform using metadata and data from the web, linked open data, harvesting and concordance rules, and bibliographic record authorities. The SMESE V3includes three distinct engines to:

1. Discover enriched sentiment/emotion metadata hidden within the text or linked to multimedia structure using the proposed BM-SSEA (BM-Semantic Sentiment and Emotion Analysis) algorithm.
2. Implement rule-based semantic metadata internal enrichment.
3. Propose a hybrid machine learning model for metadata enrichment.
4. Generate semantic topics by text, and multimedia content analysis using the proposed BM-SATD (BM-Scalable Annotation-based Topic Detection) algorithm.

The semantic aggregation of metadata content repository offers a foundation for an interest-based digital library of semantic mining activities, such as search, discover and smart notifications.

Table 1 shows the comparison with most known text mining algorithms (e.g., AlchemyAPI, DBpedia, Wikimeta, Open Calais, Bitext, AIDA, TextRazor) and a new algorithm SMESE with many attributes including keyword extraction,

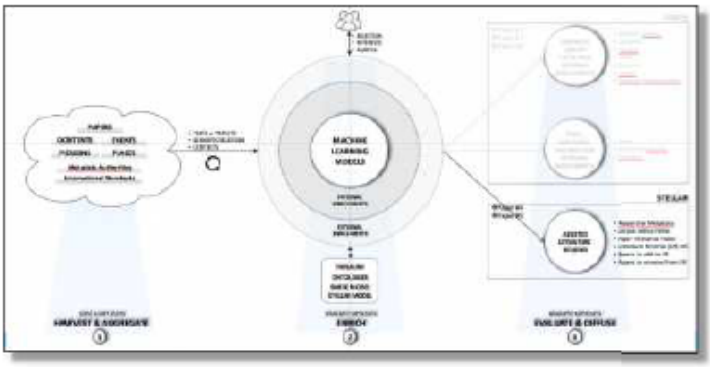


Fig. 10 Future work: Semantic Topics Ecosystem Learning-based Literature Assistant Review

concept extraction. It was noted that SMESE algorithms support more attributes than any other algorithms.

In future work, the focus will be to generate learning-based literature review enrichment and abstract of abstract.STELLAR (Semantic Topics Ecosystem Learning-based Literature Assistant Review) assess each citation to determine her ranking and her inclusion in the final literature assistant review. One goal of this enhanced ecosystem will be to reduce reading load by helping researcher to read only an intelligent selection of documents, using text data mining, machine learning, and a classification model that learn from users annotated data and detected metadata (see Fig. 10).

References

1. O. Appel, F. Chiclana, J. Carter, and H. Fujita, "A hybrid approach to the sentiment analysis problem at the sentence level," *Knowledge-Based Systems,* vol. 108, pp. 110-124, 2016. doi:http://dx.doi.org/10.1016/j.knosys.2016.05.040
2. G. Salton, and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing & Management,* vol. 24, no. 5, pp. 513-523, 1988. doi:http://dx.doi.org/10.1016/0306-4573(88)90021-0
3. T. Niu, S. Zhu, L. Pang, and A. El Saddik, "Sentiment Analysis on Multi-View Social Data," in 22nd International Conference on MultiMedia Modeling (MMM), Miami, FL, USA, 2016, pp. 15-27. doi:http://dx.doi.org/10.1007/978-3-319-27674-8_2
4. K. Bougiatiotis, and T. Giannakopoulos, "Content Representation and Similarity of Movies based on Topic Extraction from Subtitles," in Proceedings of the 9th Hellenic Conference on Artificial Intelligence, Thessaloniki, Greece, 2016, pp. 1-7. doi:http://dx.doi.org/10.1145/2903220.2903235
5. H. Sayyadi, and L. Raschid, "A Graph Analytical Approach for Topic Detection," *ACM Transactions on Internet Technology,* vol. 13, no. 2, pp. 1-23, 2013. doi:http://dx.doi.org/10.1145/2542214.2542215
6. J. L. Hurtado, A. Agarwal, and X. Zhu, "Topic discovery and future trend forecasting for texts," *Journal of Big Data,* vol. 3, no. 1, pp. 1-21, 2016. doi:http://dx.doi.org/10.1186/s40537-016-0039-2
7. D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *Journal of Machine Learning Research,* vol. 3, pp. 993-1022, 2003
8. V. Bijalwan, V. Kumar, P. Kumari, and J. Pascual, "KNN based Machine Learning Approach for Text and Document Mining," *International Journal of Database Theory and Application,* vol. 7, no. 1, pp. 61-70, 2014. doi:http://dx.doi.org/10.14257/ijdta.2014.7.1.06
9. G. A. Patel, and N. Madia, "A Survey: Ontology Based Information Retrieval For Sentiment Analysis," *International Journal of Scientific Research in Science, Engineering and Technology,* vol. 2, no. 2, pp. 460-465, 2016
10. J. A. Balazs, and J. D. Velásquez, "Opinion Mining and Information Fusion: A survey," *Information Fusion,* vol. 27, pp. 95-110, 2016. doi:http://dx.doi.org/10.1016/j.inffus.2015.06.002
11. K. Ravi, and V. Ravi, "A survey on opinion mining and sentiment analysis: Tasks, approaches and applications," *Knowledge-Based Systems,* vol. 89, pp. 14-46, 2015. doi:http://dx.doi.org/10.1016/j.knosys.2015.06.015
12. J. Serrano-Guerrero, J. A. Olivas, F. P. Romero, and E. Herrera-Viedma, "Sentiment analysis: A review and comparative analysis of web services," *Information Sciences,* vol. 311, pp. 18-38, 2015. doi:http://dx.doi.org/10.1016/j.ins.2015.03.040
13. M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Comput. Linguist.,* vol. 37, no. 2, pp. 267-307, 2011. doi:10.1162/COLI_a_00049
14. D. Vilares, M. A. Alonso, and C. GÓMez-RodrÍGuez, "A syntactic approach for opinion mining on Spanish reviews," *Natural Language Engineering,* vol. 21, no. 1, pp. 139-163, 2015. doi:http://dx.doi.org/10.1017/S1351324913000181
15. S. Kiritchenko, X. Zhu, and S. M. Mohammad, "Sentiment analysis of short informal texts," *Journal of Artificial Intelligence Research,* vol. 50, no. 1, pp. 723-762, 2014. doi:http://dx.doi.org/10.1613/jair.4272
16. S. T. Dumais, "Latent semantic analysis," *Annual Review of Information Science and Technology,* vol. 38, no. 1, pp. 188-230, 2004. doi:10.1002/aris.1440380105
17. J. Cigarrán, Á. Castellanos, and A. García-Serrano, "A step forward for Topic Detection in Twitter: An FCA-based approach," *Expert Systems with Applications,* vol. 57, pp. 21-36, 2016. doi:http://dx.doi.org/10.1016/j.eswa.2016.03.011
18. P. Chen, N. L. Zhang, T. Liu, L. K. M. Poon, and Z. Chen, "Latent Tree Models for Hierarchical Topic Detection," *arXiv preprint arXiv:1605.06650 [cs.CL],* pp. 1-44, 2016
19. R. Moraes, J. F. Valiati, and W. P. Gavião Neto, "Document-level sentiment classification: An empirical comparison between SVM and ANN," *Expert Systems with Applications,* vol. 40, no. 2, pp. 621-633, 2013. doi:http://dx.doi.org/10.1016/j.eswa.2012.07.059
20. M. Ghiassi, J. Skinner, and D. Zimbra, "Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network," *Expert Systems with Applications,* vol. 40, no. 16, pp. 6266-6282, 2013. doi:http://dx.doi.org/10.1016/j.eswa.2013.05.057
21. R. Brisebois, A. Abran, and A. Nadembega, "A Semantic Metadata Enrichment Software Ecosystem (SMESE) based on a Multi-platform Metadata Model for Digital Libraries," *Accepted for publication in Journal of Software Engineering and Applications (JSEA),* vol. 10, no. 04, 2017
22. Q. Dang, F. Gao, and Y. Zhou, "Early detection method for emerging topics based on dynamic bayesian networks in micro-blogging networks," *Expert Systems with Applications,* vol. 57, pp. 285-295, 2016. doi:http://dx.doi.org/10.1016/j.eswa.2016.03.050
23. J. M. Cotelo, F. L. Cruz, F. Enríquez, and J. A. Troyano, "Tweet categorization by combining content and structural knowledge," *Information Fusion,* vol. 31, pp. 54-64, 2016. doi:http://dx.doi.org/10.1016/j.inffus.2016.01.002

24. T. Hashimoto, T. Kuboyama, and B. Chakraborty, "Topic extraction from millions of tweets using singular value decomposition and feature selection," in 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), Hong Kong, China, 2015, pp. 1145-1150. doi:http://dx.doi.org/10.1109/APSIPA.2015.7415451
25. C. Zhang, H. Wang, L. Cao, W. Wang, and F. Xu, "A hybrid term-term relations analysis approach for topic detection," *Knowledge-Based Systems*, vol. 93, pp. 109-120, 2016. doi:http://dx.doi.org/10.1016/j.knosys.2015.11.006
26. A. A. Salatino, and E. Motta, "Detection of Embryonic Research Topics by Analysing Semantic Topic Networks," in Semantics, Analytics, Visualisation: Enhancing Scholarly Data, Montreal, Quebec, Canada, 2016, pp. 1-15
27. S. N. Shivhare, and S. Khethawat, "Emotion Detection from Text," in Second International Conference on Computer Science, Engineering and Applications (ICCSEA), Delhi, India, 2012, pp. 1-7
28. A. Moreo, M. Romero, J. L. Castro, and J. M. Zurita, "Lexicon-based Comments-oriented News Sentiment Analyzer system," *Expert Systems with Applications*, vol. 39, no. 10, pp. 9166-9180, 2012. doi:http://dx.doi.org/10.1016/j.eswa.2012.02.057
29. C. Bosco, V. Patti, and A. Bolioli, "Developing corpora for sentiment analysis: The case of irony and senti-tut," *IEEE Intelligent Systems*, vol. 28, no. 2, pp. 55-63, 2013
30. H. Cho, S. Kim, J. Lee, and J.-S. Lee, "Data-driven integration of multiple sentiment dictionaries for lexicon-based sentiment classification of product reviews," *Knowledge-Based Systems*, vol. 71, pp. 61-71, 2014. doi:http://dx.doi.org/10.1016/j.knosys.2014.06.001
31. C. Lin, Y. He, R. Everson, and S. Ruger, "Weakly Supervised Joint Sentiment-Topic Detection from Text," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 6, pp. 1134-1145, 2012. doi:http://dx.doi.org/10.1109/TKDE.2011.48
32. E. Kontopoulos, C. Berberidis, T. Dergiades, and N. Bassiliades, "Ontology-based sentiment analysis of twitter posts," *Expert Systems with Applications*, vol. 40, no. 10, pp. 4065-4074, 2013. doi:http://dx.doi.org/10.1016/j.eswa.2013.01.001
33. B. Desmet, and V. Hoste, "Emotion detection in suicide notes," *Expert Systems with Applications*, vol. 40, no. 16, pp. 6351-6358, 2013. doi:http: //dx. doi. org/ 10. 1016/j.eswa.2013.05.050
34. M. Abdul-Mageed, M. Diab, and S. Kübler, "SAMAR: Subjectivity and sentiment analysis for Arabic social media," *Computer Speech & Language*, vol. 28, no. 1, pp. 20-37, 2014. doi:http://dx.doi.org/10.1016/j.csl.2013.03.001
35. L. K.-W. Tan, J.-C. Na, Y.-L. Theng, and K. Chang, "Phrase-Level Sentiment Polarity Classification Using Rule-Based Typed Dependencies and Additional Complex Phrases Consideration," *Journal of Computer Science and Technology*, vol. 27, no. 3, pp. 650-666, 2012. doi:http://dx.doi.org/10.1007/s11390-012-1251-y
36. L. Chen, L. Qi, and F. Wang, "Comparison of feature-level learning methods for mining online consumer reviews," *Expert Systems with Applications*, vol. 39, no. 10, pp. 9588-9601, 2012. doi:http://dx.doi.org/10.1016/j.eswa.2012.02.158
37. C. Quan, and F. Ren, "Unsupervised product feature extraction for feature-oriented opinion determination," *Information Sciences*, vol. 272, pp. 16-28, 2014. doi:http://dx.doi.org/10.1016/j.ins.2014.02.063
38. S. Poria, E. Cambria, A. Hussain, and G.-B. Huang, "Towards an intelligent framework for multimodal affective data analysis," *Neural Networks*, vol. 63, pp. 104-116, 2015. doi:http://dx.doi.org/10.1016/j.neunet.2014.10.005
39. M. D. Munezero, C. S. Montero, E. Sutinen, and J. Pajunen, "Are They Different? Affect, Feeling, Emotion, Sentiment, and Opinion Detection in Text," *IEEE Transactions on Affective Computing*, vol. 5, no. 2, pp. 101-111, 2014. doi:http://dx.doi.org/10.1109/TAFFC.2014.2317187
40. W. Li, and H. Xu, "Text-based emotion classification using emotion cause extraction," *Expert Systems with Applications*, vol. 41, no. 4, Part 2, pp. 1742-1749, 2014. doi:http://dx.doi.org/10.1016/j.eswa.2013.08.073
41. V. Anusha, and B. Sandhya, "A Learning Based Emotion Classifier with Semantic Text Processing," *Advances in Intelligent Informatics*, M. E.-S. El-Alfy, M. S. Thampi, H. Takagi, S. Piramuthu and T. Hanne, eds., pp. 371-382, Cham, Switzerland: Springer International Publishing, 2015. doi:http://dx.doi.org/10.1007/978-3-319-11218-3_34
42. E. Cambria, P. Gastaldo, F. Bisio, and R. Zunino, "An ELM-based model for affective analogical reasoning," *Neurocomputing*, vol. 149, Part A, pp. 443-455, 2015. doi:http://dx.doi.org/10.1016/j.neucom.2014.01.064
43. M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130-137, 1980. doi:doi:10.1108/eb046814
44. D. M. Blei, and J. D. Lafferty, "Correlated Topic Models," in Proceedings of the 19th Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, 2005, pp. 1-8
45. D. Andrzejewski, X. Zhu, and M. Craven, "Incorporating domain knowledge into topic modeling via Dirichlet Forest priors," in Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, Quebec, Canada, 2009, pp. 25-32. doi:http://dx.doi.org/10.1145/1553374.1553378
46. T. Pedersen, S. Patwardhan, and J. Michelizzi, "WordNet::Similarity: measuring the relatedness of concepts," in Demonstration Papers at Human Language Technology conference/North American chapter of the Association for Computational Linguistics (HLT-NAACL), Boston, Massachusetts, USA, 2004, pp. 38-41
47. Y. Tsuruoka, and J. i. Tsujii, "Bidirectional inference with the easiest-first strategy for tagging sequence data," in Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, British Columbia,

Canada, 2005, pp. 467-474. doi:10.3115/1220575.1220634

48. de Marneffe M-C, MacCartney B, and Manning CD, "Generating typed dependency parsers from phrase structure parses " in fifth international conference on language resources and evaluation, GENOA , ITALY 2006, pp. 449-54

49. S. Bao, S. Xu, L. Zhang, R. Yan, Z. Su, D. Han, and Y. Yu, "Mining Social Emotions from Affective Text," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 9, pp. 1658-1670, 2012. doi:http://dx.doi.org/10.1109/TKDE.2011.188

How to cite this article:

Ronald Brisebois *et al.*2017, A Semantic Metadata Enrichment Software Ecosystembased on Machine Learning to Analyze Topic, Sentiment and Emotions. *Int J Recent Sci Res*. 8(4), pp. 16698-16714.
DOI: http://dx.doi.org/10.24327/ijrsr.2017.0804.0200

****

Paper 6:

Efficient Scientific Research Literature Ranking Model based on Text and Data Mining Technique

Ronald Brisebois, Alain Abran, Apollinaire Nadembega, Philippe N'techobo

https://www.ijerm.com/download_data/IJERM0402035.pdf

International Journal of Engineering Research And Management (IJERM)
ISSN: 2349- 2058, Volume-04, Issue-02, February 2017

Efficient Scientific Research Literature Ranking Model based on Text and Data Mining Technique

Ronald Brisebois, Alain Abran, Apollinaire Nadembega, Philippe N'techobo

***Abstract*— With the rapidly increasing of the volume of scientific publications, find quickly the relevant papers for literature review (LR) about specific topic becomes a challenging task for researchers and students. In this vein, a new literature review assistant scheme (LRAS) (1) to evaluate scientific papers relevancy according to discipline and specific topic and (2) to find papers that match a specific research topic for LR is proposed in this work. More specifically, we propose an approach based on text and data mining (TDM) that computes paper index, called Dynamic Topic based Index (*DTb Index*), takes into account (i) venues impact, (ii) authors and their affiliated institutes impact, (iii) key findings and citations impact and (iv) papers references impact. We also implement efficient search prototype that find papers according to researcher selection parameters and his annotations. The required researcher selection parameters are (i) the main topic of his research, (ii) description of his research, (iii) the title and (iv) the keywords of the paper that he plans to provide in the context of his research and for which he needs to make a LR. Based on these parameters, the engine computes the literature corpus radius index (*LCR Index*) of each paper. The main contribution of LRAS search engine prototype is the fact that the *LCR Index* takes into account the area of research. We evaluated our proposed scheme and the simulation results show that the proposed scheme outperforms traditional schemes.**

***Index Terms*—Research publications ranking, Bibliometrics, Scientometrics, Information Retrieval, Scientific literature evaluation, Reference analysis.**

I. INTRODUCTION

Literature review (LR) is one of the most important phases of research. Researchers must identify the limits and challenges about certain scientific domain. The problem is where to find the best and most relevant papers that guarantees to ascertain the state of the art on that specific domain. Certainly, the volume growth of scientific papers and the online availability of repositories allow researchers to discover, analyze and maintain an updated bibliography for specific research fields. However, in recent years, the crescent volume of scientific papers available is becoming a problem for researchers, who, unable to exploit the whole literature in a specific domain tend to follow ad-hoc approaches. In order to help researchers for the LR tasks, it becomes necessary to analyse a large volume of papers in a fairly short time. To do so, we need to evaluate paper relevance according to the scientific research domain and topic; this task refers to the ranking process of scientific papers. Ranking the relevance of scientific papers is an ongoing and a long-standing challenge.

Unfortunately, all the works about the scientific research impact are focused on the researchers ranking; however, a researcher impact is useful to rank scientific papers that he proposes. Some online academic search engines have already implemented several indices to evaluate the scientific impact of researchers, that is the case of the h-index and i10-index used in Google Scholar for evaluating researchers' impact. Most existing researchers' indexes computation algorithms are based on the number of citations received by each paper written by a researcher. For example, if a researcher has published more papers with more citations, the researcher's h-index increases. According to [1], there are four factors by which it is possible to measure the validity of scientific research: (1) number of papers, (2) impact factor of the journal, (3) the number and order of authors and (4) citations number. The number of papers speaks more about productivity than about quality while impact factor represents simple quantification of the data for scientific production. Citation analysis identifies the types of citations and measures the number of citations, self- citations. While peer-review and citation-based bibliometrics indicators have become global means of measuring research output and are playing a critical role in this process. However, citations have been criticized for limiting their scope within academic and neglecting the broader societal impact of research. Using these four factors, ranking the relevance of scientific papers cannot be done without text and data mining (TDM).

TDM can be defined as automated processing of large amounts of structured digital textual content, for purposes of information retrieval, extraction, interpretation, and analysis. Indeed, due to the large corpora of data accumulated, automated or semi-automated analysis of their contents reveals patterns that allows establishment of fact patterns invisible to the naked eye [2]. There are many reasons researchers might want to utilize TDM methods in their research. Clark [3] suggested, due to enormous growth of the volume of literature produced, that researchers should apply text mining technique to enrich the content and to perform the systematic review of literature. Indeed, mining can improve indexing, be deployed to create relevant links, to improve the reading experience. Specifically in the context of TDM, text mining is a subfield of data mining that seeks to extract

Ronald Brisebois, École de technologie supérieure, Université du Québec, Canada, (ronald.brisebois.1@ens.etsmtl.ca).
Alain Abran, École de technologie supérieure, Université du Québec, Canada, (alain.abran@etsmtl.ca).
Apollinaire Nadembega, Network Research Lab, University of Montreal, Canada, (apollinaire.nadembega@umontreal.ca).
Philippe N'techobo, École Polytechnique de Montréal, Canada, (ntechobo.edoukou-philippe-armel@polymtl.ca).

International Journal of Engineering Research And Management (IJERM)
ISSN: 2349- 2058, Volume-04, Issue-02, February 2017

valuable new information from unstructured (or semi-structured) sources. Text mining extracts information from within those documents and aggregates the extracted pieces over the entire collection of source documents to uncover or derive new information. This is the preferred view of the field that allows one to distinguish text mining from natural language processing (NLP).

TDM techniques are widely used for ranking algorithms. Ranking algorithms are defined as the procedure that search engines use to give priority to the returned results. Recent years have seen increased adoption of scientometrics techniques for assessing research impact of publications, researchers, institutions, and venues; scientometric can be defined as the science that deals with evaluation of a scientific article refers to the finding quantitative indicators (index) of the scientific research success; unfortunately, the field of scientometrics focuses on analyzing the quantitative aspects of the generation, propagation, and utilization of scientific information. Several approaches are proposed to rank scientific articles and measure the impact of research [1, 4-16]. Some approaches focused on journal ranking [15] while others focused on universities and research institution ranking [16]. Unfortunately, these approaches only consider publication-count or focused on citation analysis (citation-based approaches); the aggregate citation statistics are used to come up with evaluative metrics for measuring scientific impact. They ignored the quality of articles in term of new contribution and scientific impact and limited the evaluation to quantitative aspect. Despite several criticisms of citation-based measures for impact, it is still the subject of much scientometrics research; a highly cited paper for a given scientific research field has influenced many other researchers; new contribution includes methods for evaluating research institutions, journals and researchers. Indeed, the main approach for scientific articles ranking is the citation analysis that is mainly the number of time that a paper is cited; unfortunately, traditional approach does not consider the publisher, conference or workshop relevance. In addition, the social aspect is not taking into account; indeed, the peers' evaluations need to be considered to measure the quality of an article; the opinion of the scientific community of the research field may contribute to identify the relevant articles. Most of these approaches reduce a citation to a single edge between the citing and cited paper and treat all the edges equally. This is clearly an oversimplification since all citations are not equal and need to be considered distinctly.

According to Wan and Liu [17], as a simple extension, taking into account the number of times a paper is cited in the citing paper often does a better job of measuring the impact of the cited paper; in order word, citations should be consider to evaluate papers impact. The text around a citation anchor can be used to assess the attitude of the citing paper towards the cited paper; for example, the citation category may define citing paper attitude. And aggregating the attitudes of all the citations to a paper can give us a quantitative measure of the attitude of the community towards that paper. However, in addition to citations, others aspects need to be consider such as: (1) analyzing of social aspects of scientific research, (2) analyzing history, (3) structure and progress of scientific fields and (4) measuring inter-disciplinary of scientific fields.

For example, the ranking of scientific journals is important because of the signal it sends to scientists about what is considered most vital for scientific progress. Journal rankings are also important because they provide a filter for researchers in the face of a rapidly growing scientific literature; they provide a way to quickly identify those articles that other researchers in a field are most likely to be familiar with.

In this paper, we propose scheme, called Literature Review Assistant Scheme (LRAS), that allows computing the ranking index of the relevance of scientific papers and subsequently, allows searching papers that best match with the researcher selection parameters. The main objective of LRAS is to assist the researchers in the LR redaction tasks that consist to, first, find papers which match with their research topic and secondly, evaluate the relevance of these papers. LRAS proposes two main processes:

1) The first process of LRAS allows evaluating the relevancy of a scientific paper for a given domain and research topic; to do that, LRAS computes the paper ranking index, called Dynamic Topic based Index (***DTb Index***) making used of TDM technique. Indeed, to compute the DTb Index, LRAS considers several criteria such as (i) venue age and impact, (ii) citation category and polarity, (iii) authors' impact, (iv) authors' institutes impact and (v) citing document of cited document. In contract to existing ranking algorithm, LRAS focuses on the paper age and author social activities in terms of researcher. Ranking algorithm also considers the number of time a paper is cited in the same documents.
2) The second process of LRAS allows finding the scientific papers that best match with the researchers' topics for their LR. Notice that the traditional search algorithms use only the titles of papers as selection parameter. In contract to them, LRAS search algorithm considers (i) the main topic of the research, (ii) description of the research, (iii) the title and (iv) the keywords of the paper that researcher plans to provide in the context of his research and for which he needs to make an LR. The LRAS search algorithm is based on TDM technique. The main contribution of LRAS search engine prototype is the fact that the algorithm takes into account the area of research.

The remainder of this paper is organized as follows. Section II presents some related work. Section III describes our proposed literature review assistant scheme (LRAS) using TDM approaches. Section IV evaluates the proposed literature review assistant scheme (LRAS) via simulations. Section V concludes this paper.

II. RELATED WORKS

The network-based analysis is a natural and common approach for evaluating the scientific credit of papers. Although the number of citations has been widely used as a metric to rank papers, recently some iterative processes such as the well-known PageRank algorithm have been applied to the citation networks to address this problem. In the context of this work, several existing approaches for scientific papers ranking [5, 6, 9-12, 14, 16-19] have been analysed.

International Journal of Engineering Research And Management (IJERM)
ISSN: 2349- 2058, Volume-04, Issue-02, February 2017

Bornmann et al. [14, 16] proposed an web application to measure the performance of research institutions. They used two indicators to perform their measurement: best paper rate and best journal rate. Best paper rate is the proportion of the institutional publications which belong to the 10% most frequently cited publications in their subject area and publication year. The best journal rate is the proportion of publications which an institution publishes in the most influential journals worldwide. According to the authors, the most influential journals are those which are ranked in the first quartile (top 25%) of their subject areas as ordered by the indicator SCImago Journal Rank (SJR).

Ranking researchers, journals and institutions may not allow to evaluate the scientific papers relevancy; however, they may be use in this scientific papers relevancy index computation. Indeed, Marx and Bornmann [12] presented an overview of methods based on cited references, and examples of some empirical results from studies are presented. according to authors, the use of a selection for the analysis of references from the publications of specific research areas should enable the possibility of measuring citation impact target-oriented (i.e. limited to these areas). They mentioned that some empirical studies have shown that the identification of publications with a high creative content seems possible via the analysis of the cited references. For authors, cited reference analysis indicate the great potential of the data source. Authors also mentioned the new method, known as citing side normalization where each individual citation receives a field-specific weighting; to compute, each citation is divided by the particular number of references in the citing work.

Wan and Liu [17] proposed citation-based analysis to evaluate scientific impact of researchers in the context of Author-Level-Metric, called WL-index. Indeed, they raised the issue of the consideration of number of time that a cited paper is mentioned in a citing paper. According to authors, the counting based on the binary citation relationships is not appropriate; in a given article, some cited references appear only once, but others appear more than once. WL-index is a variant of h-index where the number of times cited paper is mentioned is considered. Indeed, take into account the number of times a cited paper is mentioned in citing paper is good idea; unfortunately, their proposed contribution cannot allow to measures impact of paper in order to identify relevant contribution; in addition, they do not consider the category of citation to evaluation scientific impact of researchers.

Hassan et al. [6] proposed a new ranking algorithm for scientific research papers, called Paper Time Ranking Algorithm (PTRA), that depends on three factors to rank its results: *paper age, citation index and publication venue*; they gave priority to each one of these parameters. Indeed, for a given paper, they computed its weight as the sum of the age of the conference or the journal impact factors, the number of citation of the paper and the age of paper. Unfortunately, they do not consider Author-Level-Metric and ignore the citation category in the computation of their citation index. Also, considering the number of citations is not good approach due to the age of paper; indeed, newspapers are penalized; they may use the average number of citations instead on the number of citations.

Rúbio and Gulo [11] proposed recommending papers based on known classification models including the paper's content and bibliometric features. Indeed, they combined text mining efforts and bibliometric measures to automatically classify the relevant papers. They made use paper's metadata such as *year of publication, citation number, references number and type of publication (journal, conference, workshop, etc.)* to measure the paper relevancy for specific science field. In they approach, they applied a ML algorithm ID3 for papers relevancy classification based on specialist annotation. Authors mentioned that their approach combines text mining and bibliometric; unfortunately, their approach only used bibliometric metrics. However, making use of machine learning (ML) technique is good things.

Madani and Weber [5] proposed an approach that applied bibliometrics analysis and keyword-based network analysis to recognize the main papers, authors, universities, and journals. Indeed, they made use bibliometrics (quantitative approach) analysis to find a general view about top authors, journals, universities, and countries; to find the most effective papers, they applied the 'eigenvector centrality' measure. For the patent evaluation, they extracted keywords from abstracts, created keyword-based network that is analyzed by cluster analysis to find groups of keywords making use of minimum spanning tree method. The list of limitations is: (1) authors do not explain how the keyword-based network is build; (2) they use only existing method and approach; and (3) paper manual annotated keywords (those given by authors of papers) are better than extracted keywords.

Wang *et al* [10] proposed a unified ranking model of scientific literature, called *MRFRank*, that employed the mutual reinforcement relationships across networks of papers, authors and text features. More specifically, *MRFRank* is proposed by incorporating the extracted text features and constructed weighted graphs. Indeed, for the same sentence, they extracted words and words co-occurrence form title and abstract. Then, they computed the TF-IDF of each word as the weight of this word. The main limitation of this approach is the fact that authors just consider the abstract to compute the weight of the word.

Gulo et al. [18] proposed a solution that automatically classifies and prioritizes the relevance of scientific papers; the solution combined text mining and ML techniques as support to identify the most relevant literature. According to authors, their approach allows to browse huge article collections and quickly find the appropriate publications of particular interest by using ML techniques. Indeed, based on previous samples manually classified by domain experts, they applied a Naive Bayes Classifier to get predicted articles; a human expert in a specific domain has analyzed each one of the training set of publications and classified the priority of the references regarding two main criteria: relevance of the reference and adequacy to the interested scientific domain. Then, based on the outputs of experts, the process of automatic classifying publications starts with a selected set of keywords that represent the context and the area of interest. As the entire supervised learning algorithm, manual contribution is highly required.

To conclude, we summarize the limitations of existing approaches for ranking the relevance of scientific papers as

International Journal of Engineering Research And Management (IJERM)
ISSN: 2349- 2058, Volume-04, Issue-02, February 2017

follows:

1) they only use citations count; in addition, they do not consider the age of papers, penalizing the recent papers;
2) they do not consider the category and polarity of citations;
3) they do not consider the other types of venues, such as conferences and workshops. In addition, what about unpublished documents?
4) for those which are based on machine learning technique; they require a large manual contribution of specialists or experts for the training step of the learning model;
5) for those which are based on text analysis to identify relevant papers; they are limiting themselves to title and abstract.

In this paper, we propose a scheme that proposes solutions to overcome these limitations. The proposed LRAS considers several criteria such as venue age and impact, citation category and polarity, authors' impact, authors' institutes impact and citing document of cited document.

III. LRAS: Literature Review Assistant Scheme

Here, we present the details of the proposed scheme, called LRAS. More specifically, we present (A) the TDM process used by LRAS to compute the relevance ranking index that denotes the relevancy of a scientific paper for a research topic and (B) the TDM based process used by LRAS to find best papers for literature review (LR) of specific research topic.

A. *Dynamic Topic based Index (DTb Index) computation process*

As mentioned above, most of existing ranking approaches focus on measuring the influence of a scientific paper based on the citations analysis. In contract to these approaches, LRAS computes the DTb Index that denotes the paper relevancy according to a specific research domain and topic; that is why this index is called dynamic topic based.

More specifically, the DTb index is also computed as a weighted sum of the values that denote the importance of the different inputs considered. The DTb index is computed using a number of additional features:

1) Key findings and peer citations index (see equation 1),
2) Venue index (see equation 2 to 6),
3) Document references index (see equation 7 to 8),
4) Authors and their affiliated institutes (see equation 9 to 12).

In contrast to existing ranking approaches, the LRAS is not limited to journal-level metrics; it also considers conference proceedings and workshop metrics; making LRAS, a scheme based also on venue-level metric.

In the rest of this section, we show how the different concepts are used to compute the DTb Index (see equation 13).

1) *Paper relevance according to researchers' key findings and peer citations*

The Key Findings are the annotations in regards to important findings in the paper. Indeed, previous researchers who have already analyzed the paper have provided annotations called key findings. These key findings are identified and analyzed by the TDM approach. The TDM analysis consists in classifying the key findings into three categories:

1) *Very relevant:* indicates that the paper is very relevant and adequate for the LR,
2) *Adequate:* indicates that the paper is not relevant, but may be the focus of attention, if possible.
3) *Not relevant:* indicates that the paper is not relevant and not adequate for the search.

Let:

1) *Cat_annot* be the category of a key finding;
2) *Y* be the age of a paper *d;*
3) *X* be the publication date of *d*;

For example: for a paper published in 2000, Y =16 and X=2000.

The key findings index of paper *d* is computed as follows:

$$KeyFindingsIndex(d, Cat\_Annot, Y) = \frac{\sum_{i=0}^{Y-1}\left((Y-i)\times Nb(d, Cat\_Annot, (X+Y-i))\right)}{Y!} \quad (1)$$

where Nb(d,Cat_Annot,Z) denotes the number of times the key findings Cat_Annot= "very relevant" are detected in paper *d* at year Z.

The concept behind the computation of the key findings index is to give more importance to the more recent annotations instead of simply counting the number of considered key findings. This places more emphasis on recently published papers.

2) *Paper relevance according to venue*

The venue type is important in the ranking of scientific papers. The intent is to consider not only papers from academic journals, but also papers from other types of venues, such as conference proceedings and workshops, as well as unpublished papers such as research reports. In LRAS, four types of venue are considered:

1) Journal
2) Conference proceedings
3) Workshop
4) Unpublished.

Here, the venue types are ordered according to their importance in the researcher's opinion. For example: a researcher may consider that a journal paper is more important than a conference proceedings paper; thus, journal is first and conference is second. To compute the venue impact, LRAS evaluates the similarity between (1) the venue topic and the papers main topic and (2) the venue name and the papers title. The similarity matching of the paper's main topic (we assumed that the research topic of the paper is known in advance) with the venue's topics (where paper *d* is published or presented) is computed as follows:

$$sim\_topic(Td, Tv) = \max_{j\in[1,m]}\left(j - gram(Td, Tv)\right) \quad (2)$$

where *Td* and *Tv* denote the main topic of paper *d* and the main topic of venue *v*, respectively.

International Journal of Engineering Research And Management (IJERM)
ISSN: 2349- 2058, Volume-04, Issue-02, February 2017

The similarity matching between paper title and venue name (where paper d is published or presented) is computed as follows:

$$sim\_name(Nd, Nv) = \max_{j \in [1,m]} (j - gram(Nd, Nv)) \quad (3)$$

where Nd and Nv denote the title of document d and the name of venue v, respectively.

Thus, the venue v impact for a specific paper d is given by:

$$VenueImpact(d, v) = age\_venue(v) + avg\_num\_pub(v) + rev\_num(v) + \frac{avg_{sub(v)}}{avg_{acc(v)}} + freq(v) + sim\_topic(Td, Tv) + sim\_name(Nd, Nv) \quad (4)$$

where

- $age\_venue(v)$ denotes the age of venue v,
- $avg\_num\_pu(v)$ denotes the number of publications per year,
- $rev\_num(v)$ denotes the number of reviewers per submitted paper,
- $avg\_sub(v)$ denotes the average number of submitted papers per year,
- $avg\_acc(v)$ denotes the average number of accepted papers per year,
- $freq(v)$ denotes the frequency of publication per year.

To take into account the type of venue, a weight is assigned to each of them according to its order and the couple (Vinit, Vunit), where:

- Vinit is an initial value and
- Vunit is the difference in weight between two consecutive types of venue.

For example: a venue type with order i will have the weight:

$$VtypeWeight(v) = Vinit + ((Q + 1 - i) \times Vunit) \quad (5)$$

where Q is the number of types of venue. Here, Q is equal to 4.

Finally, the venue-based index of paper d is computed as follows:

$$VenueIndex(d, v) = VtypeWeight(v) \times VenueImpact(d, v) \quad (6)$$

3) Paper relevance according to authors and their affiliated institutes

Until now, a number of different indicators have been proposed for evaluating the scientific impact of a scientist or a researcher, most of which are variants and revisions of h-index. However, h-index is limited to number of citations without considering the author's social personality in terms of peer award, for example. As was done for the venue index, LARS computes the paper relevance based on the authors and their affiliated institutes.

Let:

1) Td be the main topic of paper d; we assumed that the research topic of the paper is known in advance;
2) a_i be an author.

The author a_i influence on the relevance of paper d is computed as follows:

$$AuthorImpact(d, a_i) = \frac{nb\_cited(Td)}{nb\_pub(Td)} + \frac{nb\_jour(Td)}{nb\_pub(Td)} \quad (7)$$
$$nb\_award(Td, a_i) + nb\_jour(Td, I_i)$$
$$nb\_award(Td, I_i)$$

where:

- $nb\_cited(Td)$ denotes the number of publications of author a_i cited on the topic Td,
- $nb\_pub(Td)$ denotes the number of publications of a_i on the topic Td,
- $nb\_jour(Td)$ denotes the number of journal publications by a_i on the topic Td,
- $nb\_awar(Td,a_i)$ denotes the number of awards of a_i on the topic Td,
- $nb\_jour(Td,I_i)$ denotes the number of publications which a_i's affiliated institute publishes in the most influential journals worldwide on the topic Td,
- $nb\_awar(Td,I_i)$ denotes the number of awards of a_i's affiliated institute on the topic Td.

The author index for paper d is computed as follows:

$$AuthorIndex(d) = \frac{\sum_{i=1}^{A} (A + 1 - i) \times AuthorImpact(d, a_i)}{A!} \quad (8)$$

where A denotes the number of authors of paper d. The idea is to give more importance to top authors; the first author therefore has greater weight than the second author.

4) Paper relevance according to document references

The paper's interaction with other papers on the topic is measured. Two groups of papers are defined: Citing documents and Cited documents.

For a better understanding, let d be a considered paper; a citing document is a document that cited the document d, while a cited document is a document cited by the paper d. Note that the number of cited documents is static while the number of citing documents may increase with time. These two terms are important for the evaluation of document relevance. Fig. 1 illustrates the two terms according to the publication date.

International Journal of Engineering Research And Management (IJERM)
ISSN: 2349- 2058, Volume-04, Issue-02, February 2017

Efficient Scientific Research Literature Ranking Model based on Text and Data Mining Technique

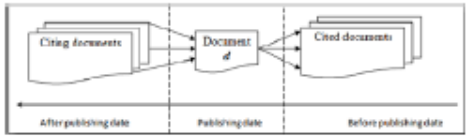


Fig. 1: Illustration of a paper reference documents

The paper's relevance based on citations includes three operands; the computation of paper's relevancy according to the references is based on the assumptions that (1) relevant papers very often cite relevant papers and (2) relevant papers are those that are frequently cited.

- Number of citing documents of paper *d* according to its age; it is computed as follows:

$$CitingImpact(d) = \frac{\sum_{i=0}^{Y-1}(Y-i)\times nb\_citing(i+1)}{Y!} \quad (9)$$

where *nb_citing(i)* denote the number of citing documents with age *i* and *Y* denotes the age of the document *d*. In addition, *CitingImpact (d)* gives more importance to recent citations.

- Average number of times a paper *d* is mentioned in citing documents; it is computed as follows:

$$CitingAvgImpact(d) = \frac{\sum_{j=1}^{P} nb\_time\_citing(d, D_j)}{P \times Y} \quad (10)$$

where $nb\_time\_citing(d,D_j)$, denotes the number of times the document *d* is cited in the citing document D_j, *P* is the total number of documents citing *d* and *Y* is the age of the document *d*.

- Number of citing documents of paper D_l (a cited document of paper *d*) according to the paper D_l age; it is computed as follows:

$$CitedCitingAvgImpact(d) = \left| \bigcup_{D_l \in L} \left\{ \frac{nb\_citing(D_l)}{age(D_l)} \geq 5 \right\} \right| \quad (11)$$

where *L* denotes the set of documents cited in *d*, $age(D_l)$ denotes the age of document D_l and $nb\_citing(D_l)$ denotes the number of times document D_l is cited.

Finally, the relevancy of paper *d* based on references is computed as follows:

$$ReferencesIndex(d) = CitedCitingAvgImpact(d) + CitingAvgImpact(d) + CitingImpact(d) \quad (12)$$

5) DTb index computation based on the previous computed index

As mentioned above, the DTb index is a weighted sum of the computed values for different features that impact the relevance of a paper.

Let the couple (Init, Unit) where:

- Init is an initial value, and
- Unit is the difference in weight between two consecutive aspects.

Init and Unit allow to assign different importance to each features. The DTb index of paper ***d*** is computed as follows:

$$DTb\,Index(d, RF, VN, AA, KF) = \frac{Val(RF,d)+Val(VN,d)+Val(AA,d)+Val(KF,d)}{\sum_{k=0}^{3}((Init+(Unit\times k)))} \quad (13)$$

where

$$Val(RF,d) = Init \times ReferenceIndex(d)$$

$$Val(VN,d) = (Init+(Unit\times 1))\times VenueIndex(d)$$

$$Val(AA,d) = (Init+(Unit\times 2))\times AuthorIndex(d)$$

$$Val(KF,d) = (Init+(Unit\times 3))\times KeyFindingsIndex(d, Cur\_Annot, Y)$$

B. Papers corpus for literature review selection process

To identify an LR corpus, the selection parameters are classified into three categories (see Table 1):

1. Evaluation-based
2. Selection-based
3. Sort-based.

Table 1: STELLAR classification of selection parameters

Evaluation-based	Selection-based	Sort-based
Main Topic (MaT) Keywords (KeW)	Discipline Languages	MLTC (Yrs, %) Number of References (<=)
Title (TiT)	LCR Index Threshold	Researcher Annotations (RA)
Description (DeC)		

Each class of the selection parameters is used for specific step on the selection process.

Selection based parameters are used to filter the papers repository in order to reduce the number of papers for the next steps; that allow to save computation cost. Sort based parameters are used to select the final list of papers for LR.

Evaluation-based parameters are used to compute the literature corpus radius (LCR) index. First, the value of each evaluation-based parameter is computed by determining the similarity of each evaluation-based selection with a predefined section of the document. The similarity matching value is in the range [0,1] where 1 means the most similar while 0 means the least similar. Next, based on the similarity matching value (e.g., the predefined weight of each of them), the LCR index is computed. Fig. 2 shows the process of LR corpus selection based on researcher's selection parameters and annotations.

International Journal of Engineering Research And Management (IJERM)
ISSN: 2349- 2058, Volume-04, Issue-02, February 2017

Indeed, the first step allows selecting a preliminary corpus of papers (C_0) based on researcher discipline and language. Then, based on the evaluation-based parameters, the LCR Index of each paper of the set of preliminary corpus of papers is computed. Then, based on the LCR Index threshold, the corpus of papers (C_1) is selected; C_1 represents the subset of C_0 where the LCR Index of papers is greater or equal to LCR Index threshold. Finally, based on the sort based parameters researcher and LCR Index, LRAS identifies the final corpus of papers (C_2) that will be used for the LR; C_2 is a subset of C_1.

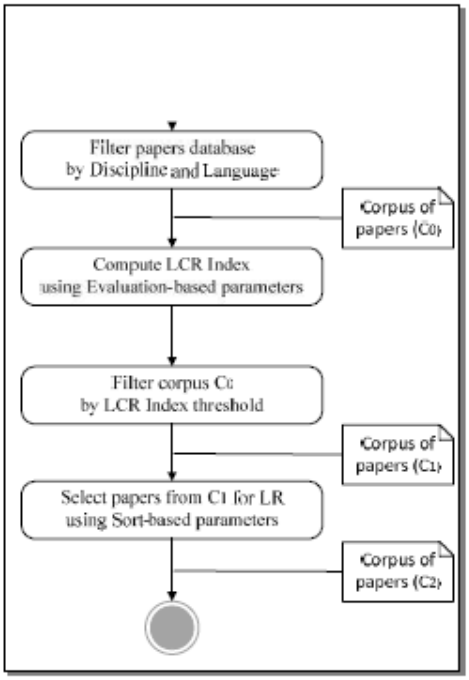


Fig. 2: Literature corpus radius (LCR) selection process

The step 1 and 3 can be performed by simply SQL request to the database using papers metadata discipline and language for step 1 and LCR Index for step 3; in the rest of this section, the details of step 2 and 4 are given.

1) Step 2 of LR corpus selection (LCRIndex computation)

As the DTb Index, LCR Index computation is based on various features that match the researcher evaluation based selection parameters. For each feature, LRAS computes the similarity matching and performs weighted sum of these similarity values to obtain the LCR Index.

For each paper, equations (14) to (16) compute the similarity of paper with the researcher's main topic while equations (17) to (18) compute the similarity of each paper with the researcher selection parameters in terms of keywords. Equations (19) to (20) compute the similarity matching of each document with the RS parameters "Title" while equations (21) to (23) compute the similarity matching of each document with the RS parameters "Description". Finally, equation (24) allows computing the LCR Index.

- *Similarity matching of a researcher main topic with the topics extracted from paper abstract*

The similarity matching with the researcher main topic is computed from the abstracts. The abstract of each is recorded in the "ABSTRACT" metadata provided by the publisher. The similarity matching computation makes use of this metadata as input to determine the paper's similarity with the researcher-defined main topic.

Let *d* be the paper and *Ad* the abstract of *d*. Next, based on the topic detection algorithm, called BM-Scalable Annotation-based Topic Detection (BM-SATD), the topics of paper *d* are detected from *Ad*; we assume that BM-SATD exists. Thus, using paper's abstract as input, BM-SATD detects their topics.

Let:

1) *Ta* be the topic detected in the abstract of paper *d*;
2) *MT* be the main topic provided as the researcher selection parameters and *n* be the number of terms of MT = (w_1, w_2, …, w_i, …, w_n);
3) *SimMatch_MaT(MT,d)* be the function that evaluates the similarity of *MT* with the paper *d* abstract; note that the terms of *MT* are ordered.

First, the i-gram of MT is calculated:

$$f(i-gram, MT, Ad) = \sum_{k=1}^{n-(i+1)} nb\left(w_k, w_{k+1}, \ldots, w_{k+i-1}\right) \quad (14)$$

where nb(w_k, w_{k+1}, …, w_{k+i-1}) is the number of times that the i-gram (w_k, w_{k+1}, …, w_{k+i-1}) appear in *Ad* (the abstract of paper *d*).

Next, the weight of the researcher's main topic for paper *d* is computed using the following equation:

$$w\_Mat(MT,d) = \sum_{i=1}^{n} i \times f(i-gram, MT, Ad) \quad (15)$$

To obtain a similarity value between 0 and 1, normalization is applied. Let *Max_MaT* be the largest value of *w_MaT(MT,d)* among all the considered papers. SimMatch_MaT(MT,d) is computed by:

$$SimMatch\_Mat(MT,d) = \frac{w\_Mat(MT,d)}{Max\_Mat} \quad (16)$$

- *Similarity matching of researcher keywords with paper keywords*

The similarity matching based on the researcher keywords is computed using the paper keywords. The keywords of each paper are recorded in the "KEYWORDS" metadata provided by the publisher.

Let:

1) *Kd* be the set of keywords of paper *d*;
2) *KW* be the set of keywords provided in the researcher selection parameters;

International Journal of Engineering Research And Management (IJERM)
ISSN: 2349- 2058, Volume-04, Issue-02, February 2017

3) *SimMatch_KeW(KW,Kd)* be the function that computes the similarity matching of *KW* with *Kd*.

First, the weight of *KW* according to paper *d* keywords *Kd* is computed as follows:

$$w\_KeW(KW,d)=|KW \cap Kd| \quad (17)$$

To obtain a similarity value between 0 and 1, normalization is applied; the SimMatch_KeW(KW,d) is computed as:

$$SimMatch\_KeW(KW,d)=\frac{w\_KeW(KW,d)}{KW} \quad (18)$$

- *Similarity matching of researcher's research title with paper title*

Before the similarity matching computation, the researcher title and paper titles are pre-processed. The objective of the pre-processing is to filter noise in order to obtain suitable text for performing the analysis. This consists in stemming, phrase extraction, part-of-speech filtering and removal of stop-words. More specifically, it includes the following operations:

1) Segmentation: the process of dividing a given document into sentences.
2) Stop-words removal: Stop-words are frequently occurring words (e.g., 'a' and 'the') that impart no meaning and generate noise. They are predefined and stored in an array. Note that the removal of stop-words follows specific rules. For example, in "prediction of mobility", removal of the stop-word "of" changes the expression to "mobility prediction".
3) Tokenization: the input text is separated into tokens.
4) Punctuation marks: the spaces and word terminators are identified and treated as word breaking characters.
5) Word stemming: each word is converted into its root form by removing its prefix and suffix for comparison with other words.

The output of the pre-processing is the set of terms.

Let:

1) *Td* be the set of terms of the title of paper *d*;
2) *TT* be the set of terms of the researcher selection title;
3) *SimMatch_TiT(TT,Td)* be the function that evaluates the similarity matching of TT with *Td*.

First, the weight of TT according to the paper *d* title *Td* is computed as follows:

$$w\_TiT(TT,d)=\max_{j\in[1,m]}\left(j-gram(TT,Td)\right) \quad (19)$$

where *m* denotes the number of terms of TT ($m=|TT|$). Indeed, w_TiT(TT,d) is the largest number of sequential terms of TT that appears in *Td*. To obtain a similarity value between 0 and 1, normalization is applied. The SimMatch_TiT(TT,d) is computed as follows:

$$SimMatch\_TiT(TT,d)=\frac{w\_TiT(TT,d)}{m} \quad (20)$$

- *Similarity matching of the researcher' research description with paper abstract*

The similarity matching of the researcher research description is performed using the paper abstract. To do this, the researcher description is semantically compared to the paper abstract in order to measure the similarity level. This similarity matching of a researcher description makes use of WordNet::Similarity, described in [20], which implements six measures of similarity and three measures of relatedness. Several terms may be semantically the same.

Let:

1) *DS* be the researcher description of the research topic as the selection;
2) *s* be the number of terms of $DS=(t_1, t_2, \ldots, t_i, \ldots, t_s)$;
3) *C* be the Literature Corpus where the papaers are of the same discipline;
4) *SimMatch_DeC(DS,d)* be the function that evaluates the similarity matching of *DS* with a paper abstract *Ad*.

First, the semantic similarity of each term in *DS* with those in *Ad* is determined on the basis of the semantic TF-ICF (term frequency – inverse corpus frequency) as follows:

$$SemSim\_T(t_i,d)=TF(t_i,d)\times\log\left(\frac{C}{ICF(t_i,C)}\right) \quad (21)$$

where C, $TF(t_i,d)$ and $ICF(t_i,C)$ denote the preliminary corpus of papers selected based on discipline and language, the number of occurrences of t_i in paper *d* and the number of papers in the corpus *C* where t_i appears.

Next, the semantic similarity of DS to the paper abstract is computed as follows:

$$SemSim\_DeC(DS,d)=\sum_{i=1}^{s}SemSim\_T(t_i,d) \quad (22)$$

To obtain a similarity value between 0 and 1, normalization is applied. The SimMatch_DeC(DS,d) is computed as:

$$SimMatch\_DeC(DS,d)=\frac{SemSim\_DeC(DS,d)}{Max\_DeC} \quad (23)$$

where *Max_DeC* denotes the largest value of SemSim_DeC(DS,d) among all the papers in *C* (i.e, preliminary corpus of papers selected based on discipline and language).

- *LCR Index computation*

Once the similarity matching of each evaluation-based selection is done, the LCR index can be computed. An LCR index value is within the range [0,1] where 0 means the least similar while 1 is the most similar. Note that the LCR index is a weighted sum of the computed value of each selection.

Let:

1) W_init be an initial value
2) W_unit be the difference in weight between two consecutive types of RS parameters.

International Journal of Engineering Research And Management (IJERM)
ISSN: 2349- 2058, Volume-04, Issue-02, February 2017

The LCR index of a paper d of literature corpus C is computed as follows:

$$LCR\ Index(d, MT, KW, TT, DS) = 1 - \left(\frac{Val(DS,d) + Val(TT,d) + Val(KW,d) + Val(MT,d)}{\sum_{i=0}^{5} (W\_init + (W\_unit \times i))} \right) \quad (24)$$

where:

$$Val(DS,d) = W\_init \times SimMatch\_DeC(DS,d)$$

$$Val(TT,d) = (W\_init + (W\_unit \times 1)) \times SimMatch\_TiT(TT,d)$$

$$Val(KW,d) = (W\_init + (W\_unit \times 2)) \times SimMatch\_KeW(KW,d)$$

$$Val(MT,d) = (W\_init + (W\_unit \times 3)) \times SimMatch\_MaT(MT,d)$$

2) Step 4 of LR corpus selecion: MLTC, Number of references and "To be included in the LR"

This sub-section describes how LRAS takes into account the researcher's requirements in terms of MLTC (Mix of the Literature Temporal Coverage (Yrs, %), number of references and the specific annotation "To be included in the LR". The MLTC allows the researcher to include a certain percentage of papers whose age is greater than a given age (Yrs).

The idea here is to be able to include very relevant papers that are out of date. To take into account both the MLTC and the number of references without prioritizing either of them, a specific algorithm is needed, which is given by the following pseudo-code. In this pseudo-code, C_1 denotes the preliminary corpus of papers selected based on discipline, language and LCR Threshold while C_2 denotes the final corpus of papers for the LR.

```
New_C1 = ∅
Old_C1 =
∅

 If (N ≤ Lengtj of All_C1)
   For the next document in All_C1
   If [(A ≠ 0) AND (B ≠ 0)]
      If [ (next document publication age ≤ y) ]
        Add next document to New_C1; A=A-1
      Else If [ (next document publication age > y) ]
              Add next document to Old_C1; B=B-1
           Else
              If [(A = 0) AND (B ≠ 0)]
                Add next document to Old_C1; B=B-1
              Else
                If [(A ≠ 0) AND (B = 0)]
                  If [ (next document publication age ≤ y) ]
                    Add next document to New_C1; A=A-1
                  Else
                    New_C1= All_C1
                    C2= New_C1 ∪ Old_C1
```

First, a list (in descending order) is created based on the LCR index applied to C_1 where the papers tagged "To be included in the LR" are at the top due to their priority; let All_C_1 be this list. Let MLTC (x, y) with its number of selections equal N: this means the researcher expects to have at most N documents, with a maximum of ***(100-x)%*** (i.e., $\frac{N}{100} \times (100 - x)$) that are at most y years old, and including all the papers tagged "To be included in the LR". Note that the latter papers have priority.

New_C_1 is defined as a sub-list of C_1 in which the paper age is less than or equal to y, and Old_C_1 contains papers older than y. Let $A = \frac{N}{100} \times x$ be the length of New_C_1 and $B = \frac{N}{100} \times (100 - x)$ be the length of Old_C_1.

Note that, when the number of papers in All_C_1 is less than N, all the documents are considered affinity matches for the LR; in that case, the MLTC selection is ignored.

However, when there are not enough papers whose age is less than or equal to y to satisfy the MLTC selection, a new MLTC is provided in order to reach the number A. But if the researcher requires the MLTC selection to be met, some papers are removed from New_C_1 in order to meet the selected MLTC(x, y).

If an "OR" has been used between the researcher sort-based selection' parameters, the LR corpus will be defined as the union of the subsets of papers provided by the MLTC process and the subsets of papers that are tagged "To be included in the LR".

Fig. 3 presents the LRAS prototype for LR corpus selection.

Fig. 3: LR corpus selection prototype

IV. PERFORMANCE EVALUATION

For the performance evaluation, we only measure the ranking relevance of papers. As comparison terms, we use the schemes described in [6] and [11], which are referred to as PTRA and ID3.

For the datasets harvest ring, LRAS prototype implements a crawler engine as [6]. This crawler consists of two main parts; automator and extractor. The main function of the automator is to retrieves search result from well-known scientific paper search engines: reseachGate, Academia, ScienceDirect, Scopus, Google scholar, Citeseerx and IEEE Xplore. The extractor extracts the useful information from the returned pages by the automator. This information's can be

International Journal of Engineering Research And Management (IJERM)
ISSN: 2349- 2058, Volume-04, Issue-02, February 2017

summarized as: the title of the paper, the abstract of paper, the year of publication, the paper citation index, the venue of publication, the venue age and type, author award, author affiliation institute and venue impact. For each paper, the downloaded bibliographic files were parsed to extract the metadata.

Unfortunately, some information does not exist, such as, the venue age and type, author award, author affiliation institute and venue impact. To solve it; first, LRAS automator used the search engines mentioned above and Google with advance search.

For the simulations, 2,000 scientific papers were used. The papers dealt with various research topics in Computer Science. Two sub-domains were chosen, each with 1,000 papers: (1) artificial intelligence and (2) information systems. In the context of these simulations, the sub-domains are treated as domains. Here, a scenario was defined as a set of two simulator runs, one on each domain dataset. For the simulator run parameters, the metadata of one paper in the dataset (discipline, language, title, topic, keywords and abstract) were used as the researcher selection parameters.

Two performance criteria were used to assess the relevancy of the papers for the researchers:

1) Accuracy: the percentage of true classifications
2) Precision: the percentage of the classified items that are relevant.

Considering the sets of relevant papers (REL) and non-relevant papers, (NREL), true relevant (TR) denotes the papers classified as REL when they really are, while false relevant (FR) denote the papers classified as REL when they are not. Thus, with the same logic, the papers classified as NREL can be true non-relevant (TN) or false non-relevant (FN). Accuracy (denoted by a) and precision (denoted by p) were computed as follows for each scenario:

$$a = \frac{TR + FR}{TR + FR + TN + FN} \qquad p = \frac{TR}{TR + FR}$$

To identify TR, FR, TN and FN for each scenario, a target paper was chosen for the domain; next, the metadata of this target paper were used as the researcher selection parameters and the references papers in the output set of the prototypes were compared to the cited papers of the target paper. Through this comparison, TR, FR, TN and FN were defined. Let $a_{i,j}$ be the accuracy of the scenario i^{th} of the dataset j and $p_{i,j}$ be the precision of the scenario i^{th} of the dataset j; the average accuracy (denoted by Avg_a_i) and the average precision (denoted by Avg_p_i) are defined as follows:

$$Avg\_a_i = \frac{\sum_{j=0}^{D} a_{i,j}}{D} \qquad Avg\_p_i = \frac{\sum_{j=0}^{D} p_{i,j}}{D}$$

where D denotes the number of datasets.

Fig. 4 shows the average accuracy for the three different scenarios (LRAS, ID3 and PTRA): the horizontal axis represents the sequence number of the simulation scenario and the vertical axis represents the average accuracy of the associated scenario. It is observed that LRAS (in red) performs better than ID3 (in green) and PTRA (in blue): LRAS has an average accuracy of 0.91 per scenario while ID3, has an average of 0.60 per scenario. The average relative improvement in accuracy (defined as [Avg_a of LRAS – Avg_a of ID3]) of LRAS in comparison to ID3 is 0.32 (32%) per scenario.

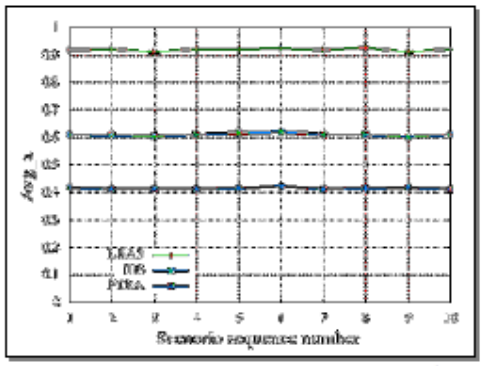


Fig. 4: Average accuracy Vs Scenario sequence number

Fig. 5 shows the average precision for the same scenarios of Fig. 4: the x Axis represents the simulations scenario sequence number while the Y axis represents the average precision of the associated scenario. LRAS performs better than ID3 and PTRA. LRAS produced an average precision of 0.96 per scenario while ID3, the best among the two works used for comparison, has an average of 0.65 per scenario. The average relative improvement in precision (defined as [Avg_p of LRAS – Avg_p of ID3]) of LRAS in comparison to ID3 is 0.31 (31%) per scenario.

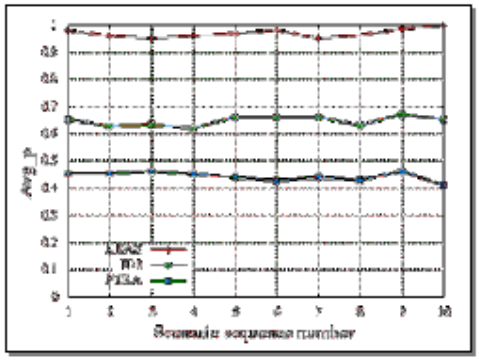


Fig. 5: Average precision Vs Scenario sequence number

V. Conclusion

In this paper, we have introduced a new scheme, which is called literature review assistant scheme (LRAS) for (1) ranking the relevance of scientific papers and (2) find the relevant papers that best match with the research topic, description and keywords of the researchers or students. More specifically, based on TDM technique, LRAS computed paper relevance index, called Dynamic Topic based Index

International Journal of Engineering Research And Management (IJERM)
ISSN: 2349- 2058, Volume-04, Issue-02, February 2017

(*DTb Index*), taking into account (i) venues impact, (ii) authors and their affiliated institutes impact, (iii) key findings and citations impact and (iv) papers references impact. To select the papers for the literature review, LRAS used the LCR Index; LRAS computed the LCR Index based on TDM technique and using (i) the main topic of his research, (ii) description of his research, (iii) the title and (iv) the keywords of the paper that he plans to provide in the context of his research and for which he needs to make a literature review. The main contribution of LRAS search engine prototype is the fact that the algorithm takes into account the area of research.

We evaluated, via simulations, LRAS and compared it against two recent related schemes proposed in [6] and [11]. The simulation results demonstrated that LRAS achieved better accuracy and precision regardless of the sequence number of the simulation scenario. For example, in comparison to ID3 proposed in [11], LRAS yielded an average relative improvement in accuracy of 32% per scenario and an average relative improvement in precision of 31%. This superior performance might be attributable to the use of additional bibliometric metadata to evaluate the relevancy of papers.

References

[1] I. MASIC, and E. BEGIC, "Evaluation of Scientific Journal Validity, It's Articles and Their Authors," *Stud Health Technol Inform.*, vol. 226, pp. 9-14, 2016.

[2] A. Okerson, "Text & Data Mining - A Librarian Overview," in 79th IFLA World Library and Information Congress, Singapore, Malaysia, 2013, pp. 1-6.

[3] J. Clark, "Text Mining and Scholarly Publishing," *Publishing Research Consortium*, 2013.

[4] M. Zhang, X. Zhang, and Y. Hu, "Ranking of Collaborative Research Teams Based on Social Network Analysis and Bibliometrics," in 12th International Conference on Cooperative Design, Visualization, and Engineering (CDVE), Mallorca, Spain, 2015, pp. 236-242.

[5] F. Madani, and C. Weber, "The evolution of patent mining: Applying bibliometrics analysis and keyword network analysis," *World Patent Information*, vol. 46, pp. 32-48, 9//, 2016.

[6] M. A. Hasson, S. F. Lu, and B. A. Hassoon, "Scientific Research Paper Ranking Algorithm PTRA: A Tradeoff between Time and Citation Network," *Applied Mechanics and Materials*, vol. 551, pp. 603-611, 2014.

[7] J. Beel, S. Langer, M. Genzmehr, B. Gipp, C. Breitinger, and A. Nurnberger, "Research paper recommender system evaluation: a quantitative literature survey," in International Workshop on Reproducibility and Replication in Recommender Systems Evaluation, Hong Kong, China, 2013, pp. 15-22.

[8] M. Cataldi, L. Di Caro, and C. Schifanella, "Ranking Researchers Through Collaboration Pattern Analysis," in European Conference on Machine Learning and Knowledge Discovery in Databases, Riva del Garda, Italy, 2016, pp. 50-54.

[9] F. Franceschini, D. Maisano, and L. Mastrogiacomo, "Influence of omitted citations on the bibliometric statistics of the major Manufacturing journals," *Scientometrics*, vol. 103, no. 3, pp. 1083-1122, 2015.

[10] S. Wang, S. Xie, X. Zhang, Z. Li, P. S. Yu, and X. Shu, "Future Influence Ranking of Scientific Literature," in Society for Industrial and Applied Mathematics (SIAM) International Conference on Data Mining, Philadelphia, Pennsylvania, USA, 2014, pp. 749-757.

[11] T. R. P. M. Rubio, and C. A. S. J. Gulo, "Enhancing Academic Literature Review through Relevance Recommendation," in 11th Iberian Conference on Information Systems and Technologies, Gran Canaria, Canary Islands, Spain, 2016, pp. 70-75.

[12] W. Marx, and L. Bornmann, "Change of perspective: bibliometrics from the point of view of cited references—a literature overview on approaches to the evaluation of cited references in bibliometrics," *Scientometrics*, vol. 109, no. 2, pp. 1397-1415, 2016.

[13] Y. Dong, R. A. Johnson, and N. V. Chawla, "Can Scientific Impact Be Predicted?," *IEEE Transactions on Big Data*, vol. 2, no. 1, pp. 18-30, 2016.

[14] L. Bornmann, M. Stefaner, F. d. M. Anegon, and R. Mutz, "Ranking and mapping of universities and research-focused institutions worldwide based on highly-cited papers: A visualisation of results from multi-level models," *Online Information Review*, vol. 38, no. 1, pp. 43-58, 2014.

[15] M. Packalen, and J. Bhattacharya, "Neophilia Ranking of Scientific Journals," *National Bureau of Economic Research Working Paper Series*, vol. 21579, 2015.

[16] L. Bornmann, M. Stefaner, F. d. M. Anegon, and R. Mutz, "Ranking and mappping of universities and research-focused institutions worldwide: The third release of excellencemapping.net," *COLLNET Journal of Scientometrics and Information Managemera*, vol. 9, no. 1, pp. 65-72, 2015/01/02, 2015.

[17] X. Wan, and F. Liu, "WL-index: Leveraging citation mention number to quantify an individual's scientific impact," *Journal of the Association for Information Science and Technology*, vol. 65, no. 12, pp. 2330-1643, 2014.

[18] C. A. S. J. Gulo, T. R. P. M. Rubio, S. Tabassum, and S. G. D. Prado, "Mining Scientific Articles Powered by Machine Learning Techniques," *OASIcs-OpenAccess Series in Informatics*, vol. 49, pp. 21-28, 2015.

[19] P. Mayr, A. Scharnhorst, B. Larsen, P. Schaer, and P. Mutschke, "Bibliometric-Enhanced Information Retrieval," in 36th European Conference on IR Research (ECIR), Amsterdam, The Netherlands, 2014, pp. 798-801.

[20] T. Pedersen, S. Patwardhan, and J. Michelizzi, "WordNet::Similarity: measuring the relatedness of concepts," in Demonstration Papers at Human Language Technology conference/North American chapter of the Association for Computational Linguistics (HLT-NAACL), Boston, Massachusetts, USA, 2004, pp. 38-41.

Paper 7:

Text and Data Mining & Machine Learning Models to Build an Assisted Literature Review with Relevant Papers

Ronald Brisebois, Alain Abran, Apollinaire Nadembega, Philippe N'techobo

http://www.ijsrise.com/index.php/IJSRISE/article/view/58/pdf

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

TEXT AND DATA MINING & MACHINE LEARNING MODELS TO BUILD AN ASSISTED LITERATURE REVIEW WITH RELEVANT PAPERS

Ronald Brisebois[1], Alain Abran[1], Apollinaire Nadembega[2*], Philippe N'techobo[3]
[1] École technologie supérieure, Université du Québec, Canada
[2] Network Research Lab., University of Montreal, Canada
[3] École Polytechnique de Montréal, Canada
*Corresponding author E-mail: apollinaire.nadembega@umontreal.ca

Abstract.

In the process of literature review writing, researchers need to search and read several papers to find those which are relevant to their research. This paper proposes an assisted literature review prototype (STELLAR – Semantic Topics Ecosystem Learning-based Literature Assistant Review) based on (1) text and data mining models that learn from researchers' annotated data and semantic enriched metadata, (2) machine learning models (MLM) and (3) a semantic metadata ecosystem (SMESE) (i) to discover papers and recommend relevant of them for a specific topic using ranking algorithm and (ii) to identify papers according to researchers' selections parameters and his annotations. Notice that SMESE is our prototype that semantically harvests papers from different sources.

Specifically, STELLAR allows to:

1. Identify the relevant papers from SMESE thanks to the computation of a new ranking index (called, DTb Index) based on paper's semantic and contextual metadata such as discipline, topic, venue, authors in order to define the Literature Corpus of a specific topic or area of research.
2. Define the Literature Corpus Radius making use of value of the similarity between each paper and a specific research area, topic, title and description (called LCR Index).
3. Assist the researcher in refining the list of papers relevant for the literature review. To narrow down the search for relevant papers, many views and relationships of the list of candidate papers are made available.

Using various types of datasets and a simulation prototypes, the STELLAR performance was evaluated and compared to two existing approaches.

Keywords: assisted literature review, literature review, machine learning, literature review enrichment, semantic topic detection, text and data mining.

1. INTRODUCTION

With the evolving, interdisciplinary and digital nature of research, there are more and more scientific publications; which increases enormously the volume of scientific papers. However, the huge volume of scientific publications available is becoming an issue for researchers (Boote & Beile, 2005; Mayr, Scharnhorst, Larsen, Schaer, & Mutschke, 2014): given that their time is limited, it is becoming impossible for researchers to read and carefully evaluate every publication within their own specialized field. Whether a short review as an assignment in a Master's program, or a LR for a PhD thesis, students find it difficult to produce a literature review (LR).

To obtain a manual LR, the researchers must dedicate to searching for literature will vary according to their research topic; which is very labor intensive. For instance, Gall et al. (Gall, Borg, & Gall, 1996) estimate that a decent LR for a dissertation takes three to six months to complete. Researchers also have to stay aware of newly published papers on related topics to produce a meaningful LR. In (Carlos & Thiago, 2015; Gulo, Rubio, Tabassum, & Prado, 2015), authors claim that an LR must address a re-

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

search question and identify primary sources and references. An ideal LR should retrieve all relevant papers for inclusion and exclude all irrelevant papers (Carlos & Thiago, 2015; Gulo et al., 2015).

In the context of scientific research, the ranking algorithms for papers evaluation are referred to as scientometrics or bibliometrics (Beel et al., 2013; Bornmann, Stefaner, Anegón, & Mutz, 2014, 2015; Cataldi, Di Caro, & Schifanella, 2016; Dong, Johnson, & Chawla, 2016; Franceschini, Maisano, & Mastrogiacomo, 2015; Hasson, Lu, & Hassoon, 2014; Madani & Weber, 2016; Marx & Bornmann, 2016; MASIC & BEGIC, 2016; Packalen & Bhattacharya, 2015; Rúbio & Gulo, 2016; Wan & Liu, 2014; Wang et al., 2014; Zhang, Zhang, & Hu, 2015). According to literature, semantic metadata can be extracted from papers using text and data mining (TDM) algorithms while machine learning models (MLM) learn from papers and researchers' annotated papers in order to identify relevant papers for a specific topic and research field.

In this view, this paper proposes a new ecosystem prototype called STELLAR (Semantic Topics Ecosystem Learning-based Literature Assistant Review), that defines and builds an assisted literature review (ALR). The ALR is designed to reduce the load of searching and reading of papers by pointing the researcher to a recommended selection of documents. To do that, STELLAR computes the ranking index, called Dynamic Topic based Index (DTb Index) that evaluates the relevancy of each harvested paper. The DTb Index allows identifying the relevant papers for a specific research area, discipline, topic, title and description. To compute the DTb Index, STELLAR makes use of paper's contextual and semantic metadata related to (1) paper's venue, (2) paper's authors and their affiliation institutes, (3) paper's references and (4) paper's citations analysis. Specifically, STELLAR papers relevance ranking algorithm considers several papers' features such as venue age, type and impact, citations category and polarity, researchers' annotated data, authors' impact and their affiliation institute. To assist the researcher, STELLAR selects the papers from SMESE, ordered according to their relevance thanks to DTb Index, for the literature corpus definition that should be use to build the literature review. The selection process takes into account the researcher's (1) researcher discipline and language, (2) researcher main topic, (3) his research title and (4) his research description. Indeed, STELLAR computes the literature corpus radius index (LCR Index) that represents the similarity between researcher's selection parameters and each paper located in SMESE. To give a visual representation, this similarity is called radius where the center of circle is the researcher's selection parameters; more a paper matches with researcher's selection parameters, more its LCR Index tends to be equal to zero and more it gets closer to the center of the circle.

Notice that the prototype of STELLAR has been implemented using our software ecosystem described in SMESE (Brisebois, Abran, & Nadembega, Unpublished results) and SMESE V3 (Brisebois, Abran, Nadembega, & N'techobo, Unpublished results). SMESE allows controlling the access of the sources and harvesting scientific papers while SMESE V3 allows enriching the harvested papers metadata in term of topics.

The remainder of this paper is organized as follows. Section 2 presents some related work while Section 3 describes the proposed ecosystem (STELLAR) multi-platform architectural model. Section 4 describes STELLAR processes to compute DTb Index and LCR Index based on MLM and TDM concepts. Section 5 evaluates the STELLAR algorithms via simulations and shows the STELLAR prototype for LCR representation. Section 6 concludes this paper and introduces the future work.

2. RELATED WORKS

The related works analysis focuses on two research sub domain of scientific assisted literature review:

i. Machine learning models

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

ii. Ranking of scientific papers

MLMs are much exploited by scientific papers relevance ranking algorithms.

2.1. Machine learning models

To extract hidden knowledge form the scientific papers, literature recommends making use of text and data mining technique. Indeed, TDM is a sub domain of artificial intelligence (AI) which uses machine learning models to perform human tasks in terms of text analysis. A MLM explores the definition and study of algorithms that can learn from and make predictions on data. In the context of TDM, MLM is used mainly for document's metadata enrichment and literature review refinement in the assisted literature review (ALR) process. For example, in the scientific text summarization, two main MLM trends are identified:

i. Supervised systems that rely on MLM algorithms trained on pre-existing document- summary pairs.
ii. Unsupervised techniques based on properties and heuristics derived from the text. The unsupervised summarization methods (He et al., 2015) are mainly based on the weight of words in sentences, as well as the sentence position in a document.

Carlos and Thiago (Carlos & Thiago, 2015) developed a supervised MLM-based solution for text mining scientific articles using the R language in "Knowledge Extraction and Machine Learning" based on social network analysis, topic models and bipartite graph approaches. Indeed, they defined a bipartite graph between documents and topics that makes use of the Latent Dirichlet Allocation topic model.

2.2. Ranking of scientific papers

Two means of quantitatively evaluating scientific research output are discussed in the literature: peer-review and citation-based bibliometrics indicators. The main limitation of citations-based approaches have been criticized for having a scope limited to academia (Marx & Bornmann, 2016).

Citation analysis is widely used to measure impact of scientific papers. Scientific paper ranking should also depend on the venue, the location of publication, the year, the author and the citation index. Some works in the field of scientific impact evaluation (Bornmann et al., 2014, 2015; Cataldi et al., 2016; Zhang et al., 2015) address the ranking of universities, institutions and research teams. For instance, M. Zhang et al. (Zhang et al., 2015) propose a method to discover and rank collaborative research teams.

For this research, many existing approaches for scientific paper ranking have been evaluated (Bornmann et al., 2014, 2015; Gulo et al., 2015; Hasson et al., 2014; Madani & Weber, 2016; Marx & Bornmann, 2016; Rúbio & Gulo, 2016; Wan & Liu, 2014; Wang et al., 2014). They suffer from a number of limitations:

i. Most existing approaches focus on the researcher index or journal index to evaluate scientific research impact, ignoring the papers index.
ii. Most only use the citations count; do not consider the age of papers.
iii. Do not take into account the Social Level Metric, and the polarity of citations.
iv. They do not consider the other types of venues, such as conference proceedings, workshops or unpublished documents.
v. Several approaches make use of MLM but with large manual contribution.

A comparison of two approaches proposed in the literature for scientific paper ranking is presented in Table 1: PTRA (Hasson et al., 2014) and ID3 (Rúbio & Gulo, 2016).

i. PTRA: Hasson et al. (Hasson et al., 2014) propose a ranking algorithm, called Paper Time Ranking Algorithm (PTRA).
ii. ID3: Rúbio and Gulo (Rúbio & Gulo, 2016) propose recommending papers based on known models, including the paper's content and bibliometric features.

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

It can be seen from Table 1 that in ranking and identifying relevant contributions, neither of these two approaches takes into account author impact, citation category, venue impact, authors' institutes or citing documents (the six rightmost columns).

Table 1. The PTRA and ID3 approaches for ranking papers

Approaches	Year of publication	Citation number	Reference	Venue type	Venue age	Authors' impact	Citation category	Venue impact	Authors' institutes	Citing document of cited document
PTRA (Hassan et al., 2014)	X	X		X						
ID3 (Robin & Qalo, 2016)	X	X	X	X						

3. STELLAR MULTI-PLATFORM ARCHITECTURAL MODEL

In this section, an overview of the STELLAR (Semantic Topics Ecosystem Learning-based Literature Assisted Review) architectural model and its prototype based on SMESE is presented. The three main processes of STELLAR are:

i. Discovery ALR
ii. Search & Refine ALR
iii. Assist & Recommend ALR.

3.1. Workflows of assisted literature reviews

An ALR process, as illustrated in Fig. 1, should allow using MLM for automated activities. In addition, it alerts the researchers about new relevant papers, or related publications. Fig. 1 shows that STELLAR assists researchers to:

i. Discover or find relevant papers for his research topic,
ii. Search or refine his search parameters,
iii. Evaluate exiting cited papers.

In the rest of this section, the STELLAR prototype is described in more detail.

3.2. Overview of the STELLAR prototype of an assisted literature review

A LR has to be systematic: it should assess each paper to determine its ranking and whether or not it is worth including in the LR. One of the aims of an ALR is to reduce the reading load by enabling the researcher to read only relevant papers. The STELLAR prototype (see Fig. 2) uses as inputs:

i. A universal research document repository (URDR) that is made possible thanks to SMESE architecture.
ii. The enriched metadata of papers such as researchers' annotations.

STELLAR MLM algorithm learns from researchers' annotated papers and the URDR papers' metadata to recommend relevant papers for a specific research field and topic.

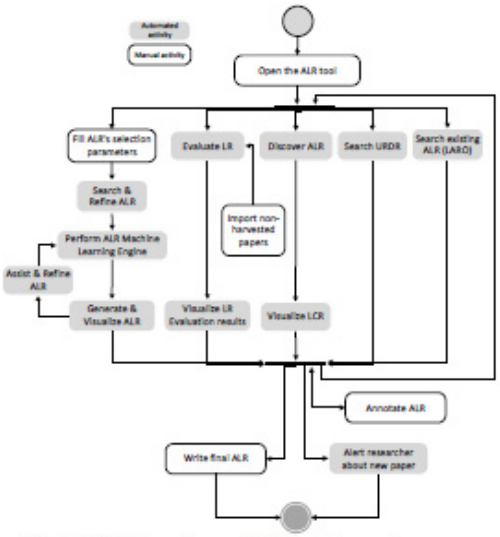


Fig. 1: Workflow of an assisted literature review

STELLAR first version prototype (STELLAR V1) architecture consists to four main parts as presented in Fig. 2:

A. Search & Refine ALR (Block A in the middle)

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

B. Assist & recommend ALR (Block B at the top-right)
C. Discover ALR Knowledge (Block C at the bottom)
D. Semantic Metadata Enrichments Software Ecosystem – SMESE V3; see (Brisebois, Abran, Nadembega, et al., Unpublished results) (top-left in Fig. 2 – see also Fig. 4).

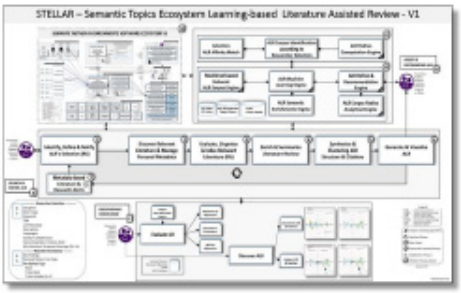


Fig. 2: STELLAR – Semantic Topics Ecosystem Learning-based Literature Assisted Review[1]

3.3. SEARCH & REFINE ALR

The Search & Refine ALR (block A in Appendix A) consists of seven steps. The first step, called **Identify, Refine & Notify ALR** allows identifying and refining the researcher selection (RS) metadata. These metadata are classified into two categories: Document Common Metadata section (top part of Table 2) and Researcher Annotations section (bottom part of Table 2). The second step is **Discover Relevant Literature & Manage Personal Metadata** that allows measuring the paper relevancy making use of the dynamic topic based index (DTb index); DTb index is computed making used of TDM approach. The third step, called **Evaluate, Organize & Index the Relevant Literature**, allows selecting the relevant papers that matches with the researcher requirement for his ALR. In contrast to Literature Corpus which denotes all the papers of a specific research topic, the ALR Corpus denotes only the papers of a Literature Corpus which meets RS metadata for an ALR. The next step, called **Enrich & Summarize the Literature Review** makes use of TDM and MLM approaches: to extract papers' subject, to detect papers' citation category en polarity, to extract papers' citation text and to performed abstract conformity. All these enrichments form the enriched metadata of paper that may be used to provide accurate summarization. **Synthesize & Clusterize the ALR Structure & Citations** step aims to synthesize and organize the relevant documents into clusters related to the LCR index while **Generate & Visualize the ALR** step aims to generate and visualize recommended papers in the Literature Corpus. Finally, **Metadata-based Literature & Research Alerts** allows detecting new relevant papers or new metadata related to the ALR.

[1] See Appendix A for a more readable version of Fig. 2.

Table 2. Researcher selection (RS) metadata

Number	Metadata	Description
	A. Document Common Metadata	
1	Discipline	Selection of the discipline related to the ALR
2	Main Topic	The main topic is one of the most important metadata for building the ALR. It should be as specific as possible.
3	Literature Corpus Radius	It is the main concept that makes it possible to refine the selection of research documents to be included in the ALR
4	Keywords	The researcher has to identify keywords representative of the ALR
5	Harvesting Date	Date of document harvesting
6	Creation Date	Date of document creation

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

7	Title	Title of the ALR
8	MLTC - Mix of Literature Temp. Cov. (Yrs, %)	MLTC is crucial to building and refining the ALR. It has two indicators: 1 - Number of years covered by the search 2 - Percentage of documents outside time-range to be included in ALR.
9	Description	A brief description of the research project of the ALR
10	Languages	The researcher has to choose the language of the papers.
11	Nb of References	The number of references that the ALR should consider.
	B. Researcher Annotations Metadata	
12	Key Findings	The Key Findings are annotations regarding important findings in the document identified by the researcher.
13	Free Tags	The researcher may place tags on a document in order to remember some information about it.
14	Personal Notes	The researcher may attach notes to a document in order to remember relative information. These notes can be used by STELLAR or the researcher to help specify the ALR.
15	Pre-defined Tags	These are predefined metadata to help the researcher. Examples: Read, To be read, To be included in the ALR

3.4. ASSIST & RECOMMEND ALR

Assist & recommend ALR (block B in Appendix A) represents the STELLR core that allows refining the ALR through two sets of steps (S1 and S2). It consists of the STELLAR MLM engines (engine 1 to 5) designed to identify a specific corpus, evaluate papers relevancy or define learning models. The Literature Corpus contains all the papers regardless of their LCR index and the type of selection metadata (i.e., RSs or RAs). The papers within corpus radius are those located at the surface (forming a disc) of a circle with the specific corpus radius – see Fig. 3.

Based on the definitions above, the Corpus Radius may be defined as the delimiter of the Literature Corpus suggested to the researcher for the ALR on the basis of the researcher's selections and annotations. The RS selection criteria are the researcher's metadata while the RA selection criteria consist of notes, tags and key findings.

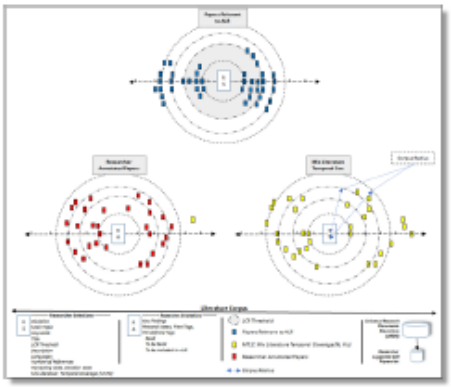


Fig. 3: Sources used to build the suggested list of ALR papers[2]

To illustrate, consider the papers in the corpus radius called ***"Papers relevant to ALR"*** (disk with blue dots at the top of Fig. 3): all the papers within the

[2] See Appendix B for a more readable version of Fig. 3

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

gray disc are whose LCR index is less than or equal to 2; in this case, the LCR threshold is set at 2.

3.5. Discover ALR Knowledge

The 'Discover ALR Knowledge' (Block C in Appendix A) has two main features. First, it allows unveiling the content of the ALR, discovering the papers harvested by SMESE and to explore the metadata generated by STELLAR MLM algorithms. Secondly, it analyses the references of manual LR in order to evaluate their relevance according to the research topic.

More specifically, the first feature "Evaluate LR" consists in an assisted evaluation of an already published LR. To evaluate an existing LR, this feature compares the existing LR to the one from STELLAR's MLM to quantify their similarity.

The tags created by the researchers are used to enrich the ALR metadata. The process 'Discover ALR Knowledge' makes it possible to drill down through different types of visualization of the corpus.

3.6. Semantic Metadata Enrichments Software Ecosystem SMESE V3

The SMESE V3 platform presented in Fig. 4 (Brisebois, Abran, Nadembega, et al., Unpublished results) is our semantic metadata enrichment software ecosystem for metadata aggregation and enrichment in order to create a semantic master metadata catalogue (SMMC). Notice that SMESE V3 includes SMESE V1 features; SMESE V3 checks continuously the access to the sources of scientific papers and analyses the data structures in order to adapt the harvesting algorithms. SMESE V3 also analyses the papers texts taking into account the documents organization and extracts the paper's research topics.

The SMESE V3 platform allows enrichment from different sources including linked open data. SMESE V3 is used by STELLAR to build its URDR (its base repository of harvested available papers at a given time *t*).

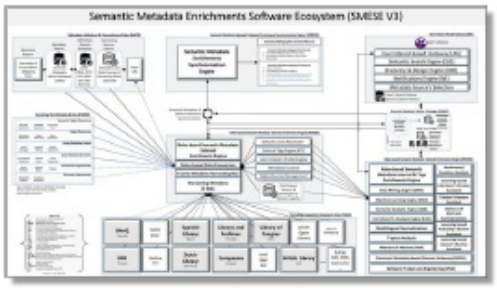


Fig. 4 SMESE V3 - Semantic Metadata Enrichments Software Ecosystem[3]

4. *STELLAR PROCESSES DESCRIPTION*

In this section, the MLM approach used by STELLAR to define its core of processes is presented. The core of STELLAR processes consists of five engines located in the bloc B (S1 and S2) of the architectural model of STELLAR. Fig. 5 shows these five engines of the core of STELLAR processes and the interaction between them to assist researchers for their ALR corpus selection. From now in this paper, the following terms are used interchangeably: document, paper and scientific paper.

Each one of these five core engines for STELLAR processes is described in detail in the following sub-sections. Indeed, using as inputs the URDR that contains existing papers, researcher annotations (RAs) and researcher selection (RS), the ***ALR radius computation engine (engine #1)*** computes the LCR index. Next, using as inputs the ALR Corpus and the training models built by researchers, ***ALR Machine Learning engine (engine #2)*** provides the ALR learning model used by the ***Multilevel-based Relevant ALR Corpus (engine #3)***. Indeed, when a new paper is harvests by SMESE, the Multilevel-based Relevant ALR Corpus of STELLAR computes the DTb Index that measures the relevancy of this paper and saves this DTb Index as new enriched metadata of the paper. The ***ALR Refine & Recommendation engine (engine #4)*** suggests the ALR references list to the researchers and assists them

[3] See Appendix C for a more readable version of Fig. 4

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

to refine this list while the ***ALR Corpus Radius Analytical engine (engine #5)*** builds dynamic graphical representations of the quantitative and qualitative metadata about selected ALR corpus.

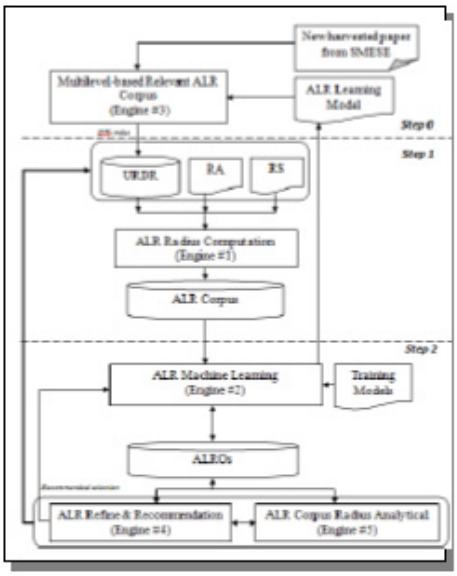


Fig. 5: Interoperability of the core engines of STELLAR processes

In the rest of the section, we focus on the first four engines.

4.1. Multilevel-based relevant ALR Corpus

The multilevel-based relevant ALR Corpus (in Step 0 and 2) is presented here. It is used to evaluate the relevancy of a paper based on a number of scientometric measurements. The measurement of relevance is referred as the ALR Index. Three types of ALR Index are defined in STELLAR: personal, collaborative and dynamic topic-based (DTb). With the personal index, the ALR corpus can be restricted to documents tagged by the researcher as "To be included in the ALR" while collaborative index restricts the ALR corpus to the documents tagged as "To be included in the ALR" by the others researchers who are selected by researcher who requests the ALR corpus. The dynamic topic-based index (DTb index) selects documents for the ALR corpus when the researcher has not requested a personal or collaborative index. The DTb index is a weighted sum of the values that denote the importance of the different inputs considered.

4.2. ALR radius computation

ALR radius computation is used to select the relevant papers to be included in the ALR, according to the researcher selection (RS) and researcher annotations (RAs). The main factor of the ALR radius computation is the LCR Index. LCR index computation is defined as a sub-algorithm of the semantic ALR selection search that identifies the ALR corpus according to the RS and RAs defined in Fig. 5; in other word, LCR Index measures the similarity between a paper, considering its text and its metadata, and the RS and RAs parameters. To identify an ALR corpus as shown in the Step 1 of Fig. 5, the selection parameters (RA and RS) are classified into three categories (see Table 3).

In the following Fig. 6, the ALR selection search using the three categories of selection parameters is explained in detail.

Table 3. STELLAR classification of researcher's selection (RS) and annotations (RAs) parameters

Evaluation-based	Selection-based	Sort-based
Main Topic	Discipline	Literature Corpus Radius (LCR)
Keywords	Languages	Mix of the Literature Temporal Coverage (MLTC)
Title	Document Researcher Annotations	Number of References
Description		

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

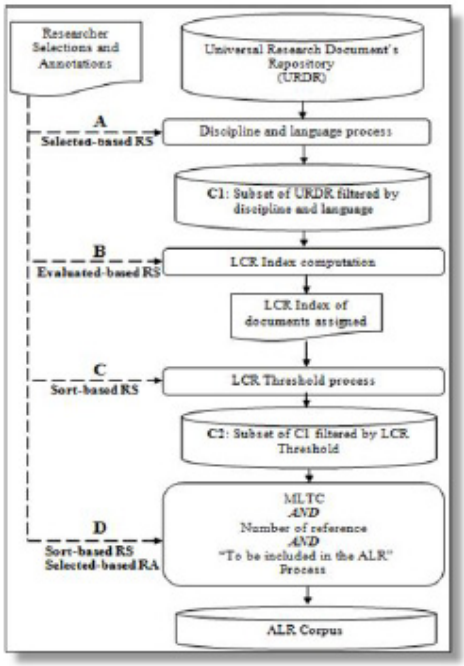


Fig. 6: Steps in a semantic ALR corpus selection search

A. Discipline and language researcher selections step

In step A in Fig. 6, volume of documents to be considered may be reduced, based on: discipline selection and language selection.

Let *DC* be the chosen discipline, let *LG* be the given language, let DISCIPLINE be the metadata that records the discipline of the documents in URDR, let LANGUAGE be the metadata that records the language of the documents in URDR and let *DiscLan_Corpus(DC,LG)* be the set of documents in the language ***LG*** that are in the discipline ***DC***. *DiscLan_Corpus(DC,LG)* is obtained as follows:

DiscLan_Corpus(DC, LG) = [select in URDR the Documents where DISCIPLINE is "DC" and LANGUAGE is "LG"]

This query to the URDR extracts only papers in the specified discipline and language. Let C_l be the corpus of papers obtained in step A.

B. LCR index computation step

Based on the set of papers selected in step A, the LCR index is computed in step B making use of the evaluation-based selections (see Table 3). The LCR index computation step consists of five sub-steps as follows:

i. Similarity matching of researcher main topic with topics extracted from document abstracts

This sub-step process, the topic detection ML model called BM-Scalable Annotation-based Topic Detection (BM-SATD) (Brisebois, Abran, Nadembega, et al., Unpublished results) is used. BM-SATD combines semantic relations between terms with co-occurrence relations across the document, by making use of the document annotations.

Here, the similarity matching is based on the n-gram approach where the value n is used as the weight (Bertin, Atanassova, Sugimoto, & Lariviere, 2016): when the i-gram expression in the researcher main topic parameter is found in the abstract, the weight *i* is associated with this expression.

Making use of the weight i_p of each paper *p* of the set C_l, the normalisation of i_p $N(i_p)$ is performed in order that $N(i_p)$ value be between 0 and 1. Let $\boldsymbol{MT_p}$ be the $N(i_p)$ of the paper *p*.

ii. Similarity matching of researcher keywords with document keywords

The weight j_p of the similarity matching of the researcher keywords parameter associated to paper *p* is the number keywords of paper *p* that are found in the set of researcher keywords parameter.

Making use of the weight j_p of each paper *p* of the set C_l, the normalisation of j_p $N(j_p)$ is performed in order that $N(j_p)$ value be between 0 and 1. Let $\boldsymbol{K_p}$ be the $N(j_p)$ of the paper *p*.

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

iii. Similarity matching of researcher title with document titles

The researcher title and papers titles are pre-processed to filter noise. This consists in stemming, phrase extraction, part-of-speech filtering and removal of stop-words. Next, based on the terms obtained, the maximum n-gram of the researcher title which is met in the paper p title is used as the title selection impact value k_p.

Making use of the value k_p of each paper p of the set C_1, the normalisation of k_p $N(k_p)$ is performed in order that $N(k_p)$ value be between 0 and 1. Let T_p be the $N(k_p)$ of the paper p.

iv. Similarity matching of researcher research topic description with document abstracts

The value l_p of The similarity matching of researcher research topic description is semantically compared with the paper p abstract using WordNet::Similarity (Pedersen, Patwardhan, & Michelizzi, 2004).

Making use of the value l_p of each paper p of the set C_1, the normalisation of l_p $N(l_p)$ is performed in order that $N(l_p)$ value be between 0 and 1. Let D_p be the $N(l_p)$ of the paper p.

v. LCR index computation

Finally, when the similarity matching of each evaluation-based selection has been completed through sub-steps 1 to 4, the LCR index within the [0,1] range can be computed. Note that the LCR index is a weighted sum of the computed value of each evaluation-based selection. The difference in weight between two consecutive evaluation-based selections (i.e., α_i and α_{i+1}) is a predefined constant value.

$$LCR\ \ Index(p) = \frac{(\alpha_1 \times MT_p) + (\alpha_2 \times K_p) + (\alpha_3 \times T_p) + (\alpha_4 \times D_p)}{\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4} \quad (1)$$

vi. Literature Corpus Radius (LCR) threshold selection step

In this step, a set of documents is sorted or selected according LCR index value. For example, a researcher may indicate that the LCR threshold is 0.7; the output will then be a subset of corpus C whose LCR index is greater than or equal to 0.7. Let C_2 be the corpus of documents obtained in step C.

vii. MLTC AND Number of references AND "To be included in the ALR" step

MLTC is the Mix Literature Temporal Coverage. Let MLTC (x, y) with its number of selections equal N: this means the researcher expects to have at most N documents, with a maximum of ***(100-x)%*** (i.e., $\frac{N}{100} \times (100 - x)$) that are at most y years old, and including all the documents tagged "To be included in the ALR". Note that the latter documents have priority.

First, a list (in descending order) is created based on the LCR index applied to corpus C_1 where the documents tagged "To be included in the ALR" are at the top due to their priority.

Let All_C_1 be this list. New_C_1 is defined as a sub-list of C_1 in which the document age is less than or equal to y, and Old_C_1 contains documents older than y.

Let $A = \frac{N}{100} \times x$ be the length of New_C_1 and $B = \frac{N}{100} \times (100 - x)$ be the length of Old_C_1. To take into account the three selections made in sub-step D.

Note that, when the number of documents in All_C_1 is less than N, all the documents are considered affinity matches for the ALR; in that case, the MLTC selection is ignored.

However, when there are not enough documents whose age is less than or equal to y to satisfy the

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

MLTC selection, a new MLTC is provided in order to reach the number *A*. But if the researcher requires the MLTC selection to be met, some documents are removed from New_C_1 in order to meet the selected MLTC(x, y).

If an "OR" has been placed between the researcher selections, the LR corpus will be defined as the union of the C2 subsets provided by the MLTC process, the Number of references process and the "To be included in the ALR" tags.

4.3. ALR Machine Learning

ALR Machine Learning (Step 2 of Fig. 5) for semantic ALR selection is the main process of STELLAR. It is a supervised MLM that makes use of a training set in order to provide the learning model.

For the rest of this sub-section, cited document denotes the paper cited by another paper while the citing document denotes the paper citing another paper.

4.3.1. Section recognition learning model

The section recognition learning model in STELLAR allows to identify each section of a paper in order to know the section of each sentence. Indeed, knowing the section in which a sentence appears may change its context. For example, citations in the 'Related Work' section do not carry the same weight as those in the 'Discussion' section in terms of identifying existing papers in a specific domain. To perform automatic section detection, manual training model is used.

4.3.2. Citations-based learning model

A citations-based learning model has been designed to identify and extract citations in documents. This learning model is divided as follows (see Table 4):

A. Citation style learning model based on citation style
B. Citation classification learning model based on rhetorical categories, cue phrases.

A cue phrase is the phrase that often occurs in a certain rhetorical category. In the case of citation classification, the verb plays the main role. Researchers are asked to read and detect the cue phrases associated with each citation polarity and category; this makes it possible to build a training model of cue phrases and their classifications, which is integrated into the "Training Model".

Table 4: Citations-based learning model

A. Citation style learning model	
Style marker	**Description**
Numerical	The syntax of this citation style is the number between brackets.
Textual	This citation style: (<names of authors>, year) or < names of authors > (year).
Personalization	This style is based on the set of texts that refer to cited papers.
B. Citation classification model	
Citation category	**Description**
Relevant	According to the citing document, the cited document is relevant.
Problem	The cited document presents the issues that led to the research.
Uses	The cited document proposes a solution that is used in the citing document.
Extension	The cited document proposes a solution that is extended by the citing document.
Comparison	The cited document proposes a solution that is compared with the citing document solution in terms of performance.

Next, based on semantic similarities, any rhetorical category that was not detected manually is detected automatically and added to the model. The

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

polarity model is proposed in order to indicate whether the citation is positive or negative.

4.3.3. Text-based learning model

To define the text-based learning model, text categories have been predefined as follows: problem, solution and results. As in the citation-based learning model, rhetorical expressions are detected by means of cue phrases. The text-based learning model is organized as follows:

1. The cue phrase learning model containing a list of cue phrases (CPs): problem CP, solution CP and result CP.
2. The thematic learning model (TRs):

a. Problem learning model: list of problem rhetorical expressions (P_TR)
b. Solution learning model: list of solution rhetorical expressions (S_TR)
c. Result learning model: list of result rhetorical expressions (R_TR).

4.4. ALR Refine & Recommendation MLM

Making use of the relevant and enriched papers identified automatically by STELLAR and contained into ALR Corpus according to the RS and RAs, the recommended selections parameters are provided to a researcher. This MLM engine recommends three different aspects of the ALR selection as shown in Fig. 7.

In other word, this engine suggests new RS parameters to the researchers in order to maximum the relevant papers for his ALR.

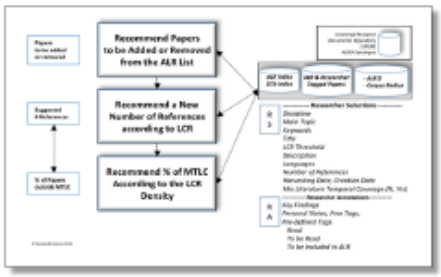


Fig. 7: Refinement & Recommendation MLM[4]

[4] See Appendix D for a more readable version of Fig. 7

5. STELLAR PERFORMANCE EVALUATION THROUGH SIMULATIONS

This section presents an evaluation of the performance of the STELLAR prototype through a number of simulations to the identification and ranking of relevant papers.

5.1. Datasets

Two datasets were used for the simulations:

i. A dataset harvested from databases
ii. A baseline dataset.

5.1.1. Dataset harvested from databases

For the simulations, 2,000 scientific papers were collected from databases such as Science Direct and Scopus. The papers dealt with various research topics in Computer Science. Two sub-domains were chosen, each with 1,000 papers: (1) Artificial Intelligence, and (2) Information Systems. For these simulations, the sub-domains are treated as domains. The other metadata were collected as bibliographic references.

For each paper, the downloaded bibliographic files were parsed to extract the metadata and were input into the SMESE V3 platform with the paper itself. Here, a scenario was defined as a set of two simulator runs, one on each domain dataset. For the simulator run parameters, the metadata of one paper in the dataset (discipline, language, title, topic, keywords and abstract) were used as the RS and RA parameters.

5.1.2. Baseline dataset

For the present study, we had already produced a manual ALR that is listed in the References section. The baseline dataset consisted of 58 papers dealing with both general and specific topics within the domain. Here, a scenario was defined as one simulator run where the 58 papers constituted the dataset. For the simulator run parameters, the metadata of the present study (discipline, language, title, topic, keywords and abstract) were used as the RS and RA parameters.

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

5.2. Performance criteria

As in (Rúbio & Gulo, 2016), two performance criteria were used to assess the relevancy of the papers for the researchers:

i. Accuracy: the percentage of true classifications
ii. Precision: the percentage of the classified items that are relevant

Considering the sets of relevant papers (REL) and non-relevant papers, (NREL), true relevant (TR) denotes the papers classified as REL when they really are, while false relevant (FR) denote the papers classified as REL when they are not. Thus, with the same logic, the papers classified as NREL can be true non-relevant (TN) or false non-relevant (FN).

Accuracy, denoted by a, was computed as follows for each scenario:

$$a = \frac{TR + FR}{TR + FR + TN + FN}$$

Precision, denoted by p, was computed as follows for each scenario:

$$p = \frac{TR}{TR + FR}$$

To identify TR, FR, TN and FN for each scenario, a target paper was chosen for the domain; next, the metadata of this target paper were used as the selection parameters and the references papers from the output set were compared to the cited papers of the target paper. Through this comparison, TR, FR, TN and FN were defined. Let $a_{i,j}$ be the accuracy of the scenario i^{th} of the dataset j; the average accuracy is defined as follows:

$$Avg\_a_i = \frac{\sum_{j=1}^{D} a_{i,j}}{D}$$

Similarly, the precision of the scenario i^{th} of the dataset j is defined as:

$$Avg\_p_i = \frac{\sum_{j=1}^{D} p_{i,j}}{D}$$

where D denotes the number of datasets.

5.3. Related ranking approaches for comparison purposes

There are two other works on scientific paper ranking:

- PTRA (Hasson et al., 2014)
- ID3 (Rúbio & Gulo, 2016).

PTRA and ID3 are described in section 2.1. Table 5 presents a summary of the criteria taken into account by each ranking approach: the bottom line of Table 5 lists all the criteria used in the STELLAR ranking approach.

Table 5. Criteria taken into account in three paper ranking approaches

Approaches	Year of publication	Citation number	Reference	Venue type	Venue age	Authors' impact	Citation category	Venue impact	Authors' institution	Citing document of cited document
PTRA (Hasson et al., 2014)	X	X		X						
ID3 (Rúbio & Gulo, 2016)	X	X	X	X						
STELLAR	X	X	X	X	X	X	X	X	X	X

The performance of the STELLAR approach was compared against the performance of PTRA (Hasson et al., 2014) and ID3 (Rúbio & Gulo, 2016) on the same datasets and scenarios. In the previous Table 7, it is observed that for ranking a cited paper as relevant, STELLAR considers more criteria, such as venue age, citation, authors' impact, etc.

5.4. Analysis of the simulation results

This section presents the analysis of the simulation results in terms of papers' relevancy for the two datasets.

5.4.1. Simulation using the dataset harvested from databases

Fig. 8 shows the average accuracy for the three different simulations (STELLAR, ID3 and PTRA). The horizontal axis represents the sequence number of the simulation scenarios and the vertical axis represents the average accuracy of the associated scenario.

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

It is observed that STELLAR performs better than ID3 (in green) and PTRA (in blue): STELLAR has an average accuracy of 0.91 per scenario while ID3 has an average of 0.60 per scenario. The average relative improvement in accuracy (defined as [Avg_a of STELLAR ¯ Avg_a of ID3]) of STELLAR in comparison to ID3 is 0.32 (32%).

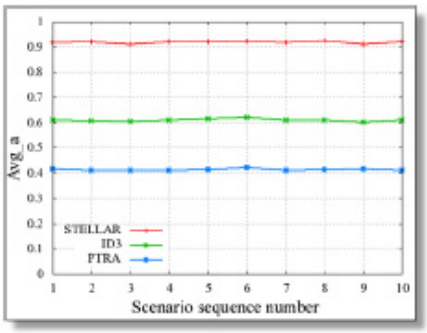


Fig. 8: Average accuracy vs Scenario sequence number – Harvested from databases

Fig. 9 shows the average precision for the same scenarios of Fig. 8. The x-axis represents the simulations scenario sequence number while the y-axis represents the average precision of the associated scenario. STELLAR performed better than ID3 and PTRA: it produced an average precision of 0.96 per scenario while ID3, the better of the two approaches used for comparison, had an average of 0.65 per scenario. The average relative improvement (defined as [Avg_p of STELLAR ¯ Avg_p of ID3]) of STELLAR in comparison to ID3 is 0.31 (31%) per scenario.

In both simulations and criteria, STELLAR outperformed ID3 and PTRA. This performance might be attributable to the use of additional bibliometric metadata.

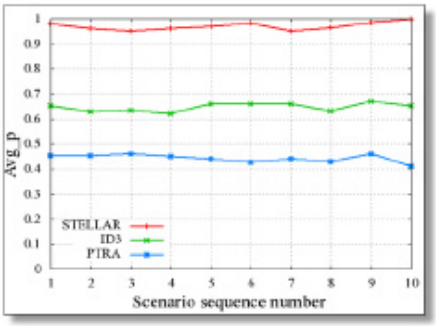


Fig. 9: Average precision vs Scenario sequence number – Harvested from databases

5.4.2. Simulation using the baseline dataset

Table 6 presents the accuracy and precision when the list of papers in the baseline dataset (i.e., the references cited in this paper) is used as the dataset for simulations with the three ranking approaches.

Table 6. Summary of performance criteria (accuracy and precision) using the baseline dataset

Approaches	Avg_a (%)	Avg_p (%)
PTRA (Hasson et al., 2014)	39.19	27.16
ID3 (Rúbio & Gulo, 2016)	53.98	41.97
STELLAR	76.09	68.73

i. STELLAR produced an average accuracy (Avg_a) of 76.09% while ID3 produced an accuracy of 53.98%. The relative improvement in accuracy of STELLAR as compared to ID3 is 22.11%.
ii. STELLAR produced an average precision (Avg_p) of 68.73% while ID3 produced a precision of 41.97%. The relative improvement in precision of STELLAR as compared to ID3 is 26.76%.

Note that all the simulations are based on limited datasets, and should be extended later to larger datasets.

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

5.5. STELLAR prototype

This section presents a number of STELLAR's input screens. It can be seen that the radius of the paper at the top of the list is 0.0: indeed, this is the target paper. Fig. 10 represents the timeline of a document-based literature corpus radius, with the horizontal axis indicating the year of publication (here, from 2011 to 2016).

Fig. 10: Timeline of a Document-based Literature Corpus Radius (LCR)

The radius denotes the temporal distance from the document at center to the cited documents and to the citing documents. The yellow circles on the left side represent multiple documents—here, 20 to 35 documents.

6. CONCLUSION AND FUTURE WORK

This paper has proposed an assisted literature review (ALR) prototype, called STELLAR (Semantic Topics Ecosystem Learning-based Literature Assistant Review). STELLAR is based on machine learning model (MLM) and a semantic metadata ecosystem (SMESE) to identify, rank and recommend relevant papers for an ALR according to researchers' selection parameters and annotations. Using text and data mining (TDM) techniques, MLM and a classification model, STELLAR assists the researcher to search relevant papers that meet his selection of parameters. The learning models applied by STELLAR use researchers' annotated (RA) data and semantic enriched metadata as training data. STELLAR also recommends selection parameters to researcher in order to refine the search.

The STELLAR prototype is based on SMESE V3, described in (Brisebois, Abran, Nadembega, et al., Unpublished results). The contributions of STELLAR include:

i. MLM designed to semantically harvest a Universal Research Documents Repository;
ii. Enhancement of Literature Corpus Radius, which compute the distance from each paper to the center of the Literature Corpus;
iii. MLM that help the researcher discover, find and refine the list of papers recommended for inclusion.

The performance of the STELLAR prototype has been evaluated through a comparison against a baseline manual LR using a number of simulations. In terms of accuracy, the STELLAR ALR provided an average accuracy of 0.91 per scenario while ID3 provided an average of 0.60 per scenario. In terms of precision, STELLAR produced an average of 0.96 per scenario while ID3 had an average of 0.65 per scenario. In comparison to ID3, STELLAR yielded an average relative improvement in accuracy of 32% per scenario and an average relative improvement in precision of 31%.

As STELLAR future work (i.e., STELLAR V2), the next contribution will focus on "Abstract of Abstracts summarization (AoA)" in order to extend STELLAR. More specifically, papers' abstracts will be used as input for our scientific paper summarization technique to generate the AoA. STELLAR V2 will allow enhancing the SMESE V3 prototype to harvest semantic metadata from more different sources as TV guides, radio channel schedule, books, music and other events calendar and create triplets to enriching metadata.

International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

REFERENCES

[1] Beel, J., Langer, S., Genzmehr, M., Gipp, B., Breitinger, C., & Nurnberger, A. (2013). *Research paper recommender system evaluation: a quantitative literature survey*. Paper presented at the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation, Hong Kong, China.

[2] Bertin, M., Atanassova, I., Sugimoto, C. R., & Lariviere, V. (2016). The linguistic patterns and rhetorical structure of citation context: an approach using n-grams. *Scientometrics, 109*(3), 1417-1434. doi:10.1007/s11192-016-2134-8

[3] Boote, D. N., & Beile, P. (2005). Scholars Before Researchers: On the Centrality of the Dissertation Literature Review in Research Preparation. *Educational Researcher, 34*(6), 3-15. doi:http://dx.doi.org/10.3102/0013189x034006003

[4] Bornmann, L., Stefaner, M., Anegón, F. d. M., & Mutz, R. (2014). Ranking and mapping of universities and research-focused institutions worldwide based on highly-cited papers: A visualisation of results from multi-level models. *Online Information Review, 38*(1), 43-58. doi:http://dx.doi.org/doi:10.1108/OIR-12-2012-0214

[5] Bornmann, L., Stefaner, M., Anegón, F. d. M., & Mutz, R. (2015). Ranking and mappping of universities and research-focused institutions worldwide: The third release of excellencemapping.net. *COLLNET Journal of Scientometrics and Information Management, 9*(1), 65-72. doi:http://dx.doi.org/10.1080/09737766.2015.1027090

[6] Brisebois, R., Abran, A., & Nadembega, A. (Unpublished results). *A Semantic Metadata Enrichment Software Ecosystem (SMESE) based on a Multi-platform Metadata Model for Digital Libraries*. International Journal for Digital Libraries.

[7] Brisebois, R., Abran, A., Nadembega, A., & N'techobo, P. (Unpublished results). *A Semantic Metadata Enrichment Software Ecosystem based on Sentiment/Emotion Analysis Enrichment (SMESE V3)*. Information Systems.

[8] Carlos, A. S. J. G., & Thiago, R. P. M. R. (2015). *Text Mining Scientific Articles using the R Language*. Paper presented at the 10th Doctoral Symposium in Informatics Engineering, Porto, Portugal.

[9] Cataldi, M., Di Caro, L., & Schifanella, C. (2016). *Ranking Researchers Through Collaboration Pattern Analysis*. Paper presented at the European Conference on Machine Learning and Knowledge Discovery in Databases, Riva del Garda, Italy. http://dx.doi.org/10.1007/978-3-319-46131-1_11

[10] Dong, Y., Johnson, R. A., & Chawla, N. V. (2016). Can Scientific Impact Be Predicted? *IEEE Transactions on Big Data, 2*(1), 18-30. doi:http://dx.doi.org/10.1109/TBDATA.2016.2521657

[11] Franceschini, F., Maisano, D., & Mastrogiacomo, L. (2015). Influence of omitted citations on the bibliometric statistics of the major Manufacturing journals. *Scientometrics, 103*(3), 1083-1122. doi:http://dx.doi.org/10.1007/s11192-015-1583-9

[12] Gall, M. D., Borg, W. R., & Gall, J. P. (1996). *Educational research: An introduction* (Vol. 6th

ed.(1996). xxii 788 pp). White Plains, NY: England: Longman.

[13] Gulo, C. A. S. J., Rubio, T. R. P. M., Tabassum, S., & Prado, S. G. D. (2015). Mining Scientific Articles Powered by Machine Learning Techniques. *OASIcs-OpenAccess Series in Informatics, 49*, 21-28. doi:http://dx.doi.org/10.4230/OASIcs.ICCSW.2015.21

[14] Hasson, M. A., Lu, S. F., & Hassoon, B. A. (2014). Scientific Research Paper Ranking Algorithm PTRA: A Tradeoff between Time and Citation Network. *Applied Mechanics and Materials, 551*, 603-611. doi:http://dx.doi.org/10.4028/www.scientific.net/AMM.551.603

[15] He, Z., Chen, C., Bu, J., Wang, C., Zhang, L., Cai, D., & He, X. (2015). Unsupervised document summarization from data reconstruction perspective. *Neurocomputing, 157*, 356-366. doi:http://dx.doi.org/10.1016/j.neucom.2014.07.046

[16] Madani, F., & Weber, C. (2016). The evolution of patent mining: Applying bibliometrics analysis and keyword network analysis. *World Patent Information, 46*, 32-48. doi:http://dx.doi.org/10.1016/j.wpi.2016.05.008

[17] Marx, W., & Bornmann, L. (2016). Change of perspective: bibliometrics from the point of view of cited references—a literature overview on approaches to the evaluation of cited references in bibliometrics. *Scientometrics, 109*(2), 1397-1415. doi:http://dx.doi.org/10.1007/s11192-016-2111-2

[18] MASIC, I., & BEGIC, E. (2016). Evaluation of Scientific Journal Validity, It's Articles and Their Authors. *Stud Health Technol Inform., 226*, 9-14. doi:http://dx.doi.org/10.3233/978-161499-664-4-93-5

[19] Mayr, P., Scharnhorst, A., Larsen, B., Schaer, P., & Mutschke, P. (2014). *Bibliometric-Enhanced Information Retrieval*. Paper presented at the 36th European Conference on IR Research (ECIR), Amsterdam, The Netherlands. http://dx.doi.org/10.1007/978-3-319-06028-6_99

[20] Packalen, M., & Bhattacharya, J. (2015). Neophilia Ranking of Scientific Journals. *National Bureau of Economic Research Working Paper Series, 21579*. doi:http://dx.doi.org/10.3386/w21579

[21] Pedersen, T., Patwardhan, S., & Michelizzi, J. (2004). *WordNet::Similarity: measuring the relatedness of concepts*. Paper presented at the Demonstration Papers at Human Language Technology conference/North American chapter of the Association for Computational Linguistics (HLT-NAACL), Boston, Massachusetts, USA.

[22] Rúbio, T. R. P. M., & Gulo, C. A. S. J. (2016). *Enhancing Academic Literature Review through Relevance Recommendation*. Paper presented at the 11th Iberian Conference on Information Systems and Technologies, Gran Canaria, Canary Islands, Spain.

[23] Wan, X., & Liu, F. (2014). WL-index: Leveraging citation mention number to quantify an individual's scientific impact. *Journal of the Association for Information Science and Technology, 65*(12), 2330-1643. doi:http://dx.doi.org/10.1002/asi.23151

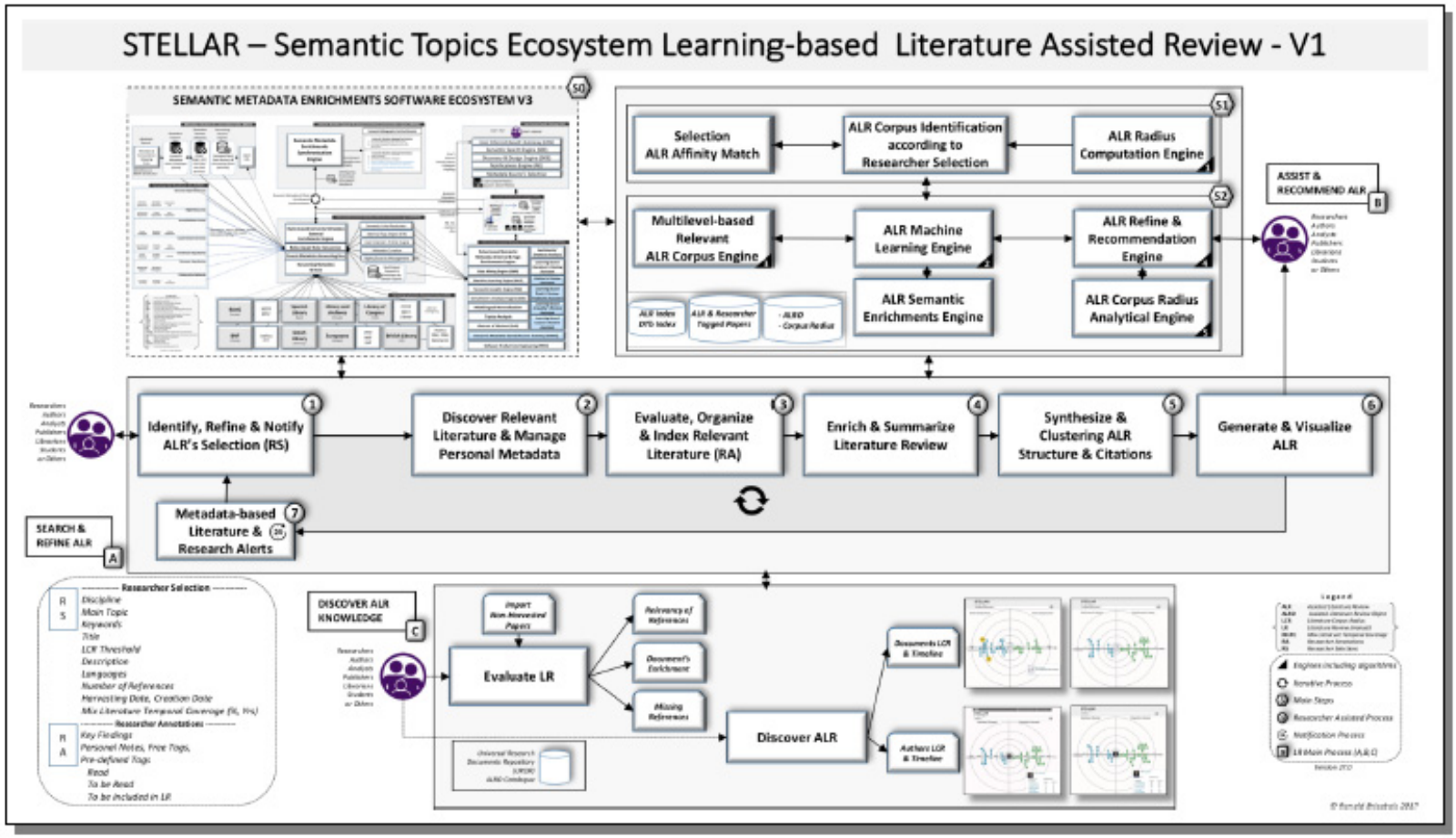
[24] Wang, S., Xie, S., Zhang, X., Li, Z., Yu, P. S., & Shu, X. (2014). *Future Influence Ranking of*

Scientific Literature. Paper presented at the Society for Industrial and Applied Mathematics (SIAM) International Conference on Data Mining, Philadelphia, Pennsylvania, USA. http://epubs.siam.org/doi/abs/10.1137/1.9781611973440.86

[25] Zhang, M., Zhang, X., & Hu, Y. (2015). *Ranking of Collaborative Research Teams Based on Social Network Analysis and Bibliometrics*. Paper presented at the 12th International Conference on Cooperative Design, Visualization, and Engineering (CDVE), Mallorca, Spain. http://dx.doi.org/10.1007/978-3-319-24132-6_30

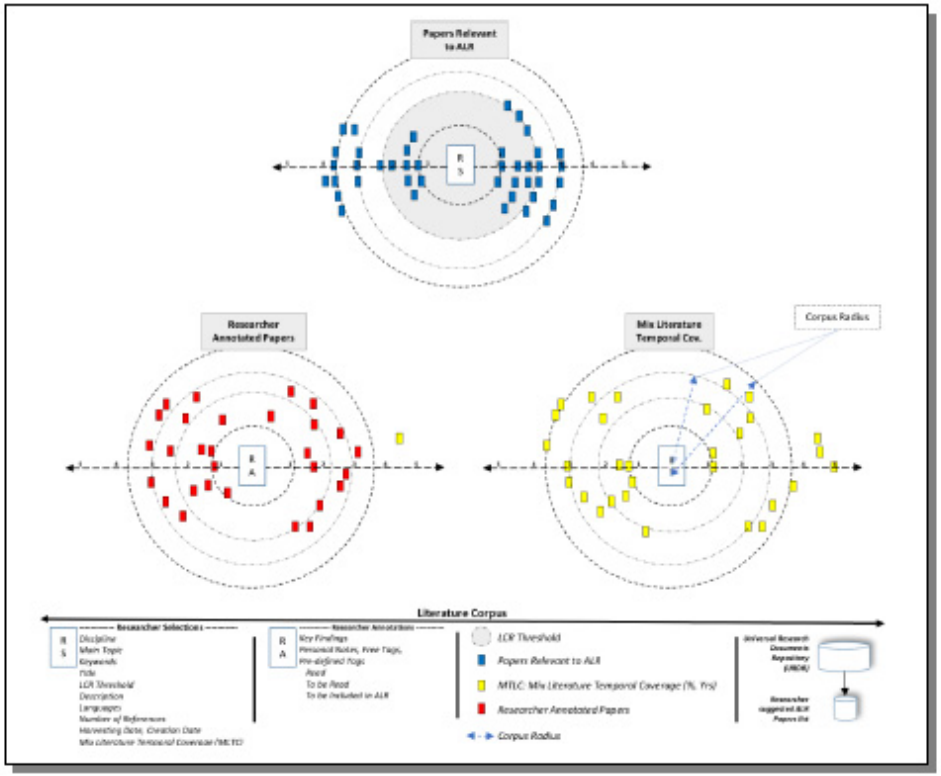
International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

Appendix A: STELLAR – Semantic Topics Ecosystem Learning-based Literature Assisted Review - V1



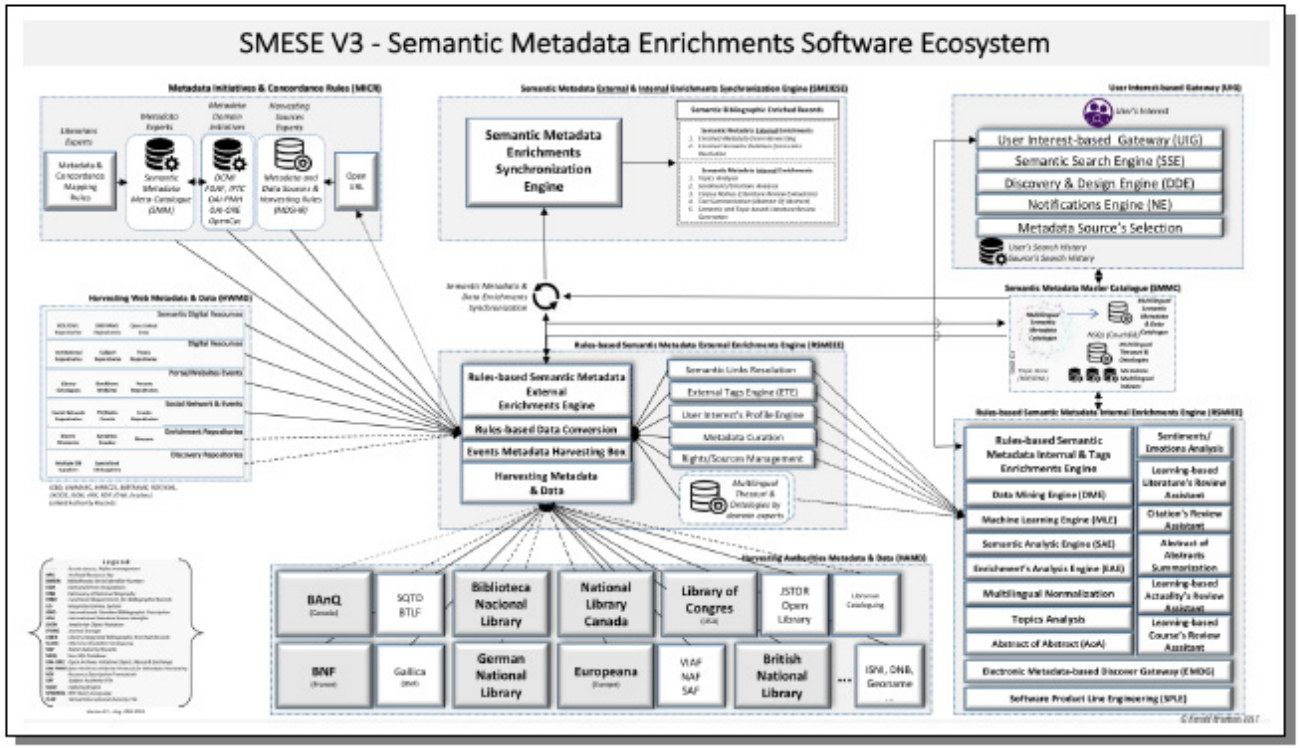
International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

Appendix B: Fig. 3 - Sources used to build the suggested list of ALR papers



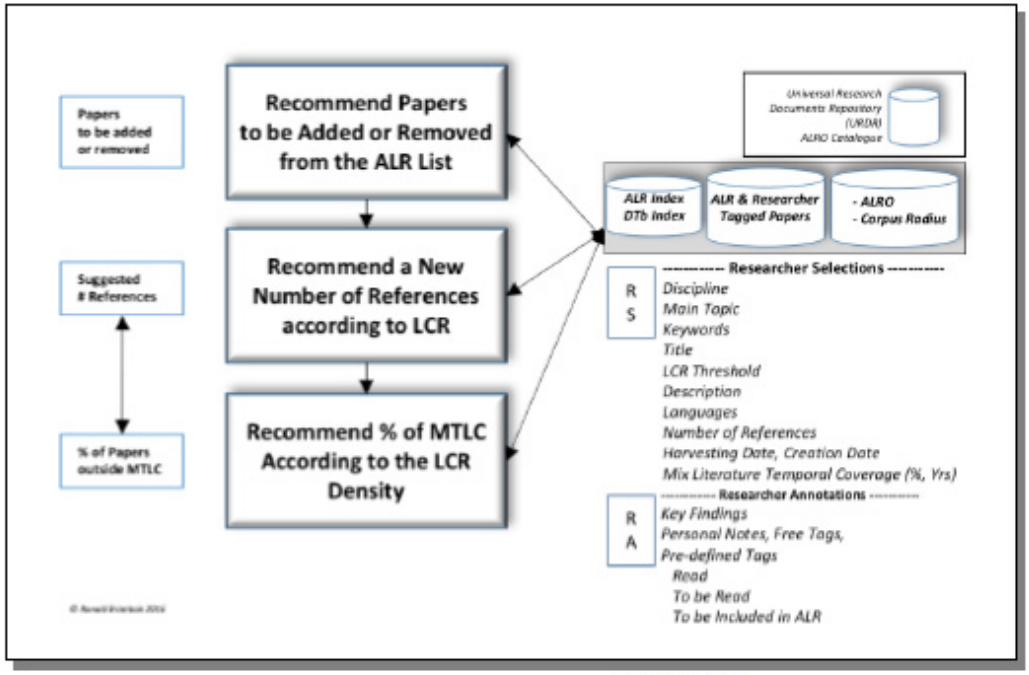
International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

Appendix C: Fig. 4 - SMESE V3 - Semantic Metadata Enrichments Software Ecosystem



International Journal of Scientific Research in Information Systems and Engineering
Volume 3, issue 1, April – 2017. ISSN 2380-8128

Appendix D: Fig. 7 - Refinement & Recommendation MLM



THESIS DEFENSE PRESENTATION

By Ronald Brisebois

ÉTS

A Semantic Metadata Enrichments Software Ecosystem (SMESE)
its Prototypes for Digital Libraries, Metadata Enrichments and Assisted Literature Reviews

Ph.D. thesis (by publication) defence
Ronald Brisebois

Thesis Supervisor
Dr. Alain Abran

Université du Québec
École
de technologie
supérieure

Montréal, May 19, 2017

1

A Semantic Metadata Enrichments Software Ecosystem
its Prototypes for Digital Libraries, Metadata Enrichments and Assisted Literature Reviews

ÉTS

1. Introduction	1. Context of the thesis (Motivations and Goals) 2. Overview of the thesis	
2. Literature Reviews	1. Software Ecosystem Model 2. Semantic Metadata Enrichments 3. Assisted Literature Reviews	
3. Major Research Themes	1. Software Ecosystem Model 2. Semantic Metadata Enrichments 3. Assisted Literature Reviews	(SMESE V1) (SMESE V3) (STELLAR V1)
4. Research Contributions	1. Published articles related to this thesis 2. Software Ecosystem Models 3. Semantic Metadata Enrichments 4. Assisted Literature Reviews	 (SMESE V1) (SMESE V3) (STELLAR V1)
5. Future Works & Questions		

2

A Semantic Metadata Enrichments Software Ecosystem

Context of the thesis

ÉTS

Research Motivations:

1. Very Limited ***Interoperability in existing Digital Library (DL)***.
2. Limited capabilities in ***Automatic Cataloguing*** (based on non-annotated metadata).
3. Limited capabilities in ***Topic, Sentiment and Emotion Extractions***.
4. Very Limited ***Assisted Literature Reviews for scientific papers.*** (no focus on researchers annotations and research metadata).

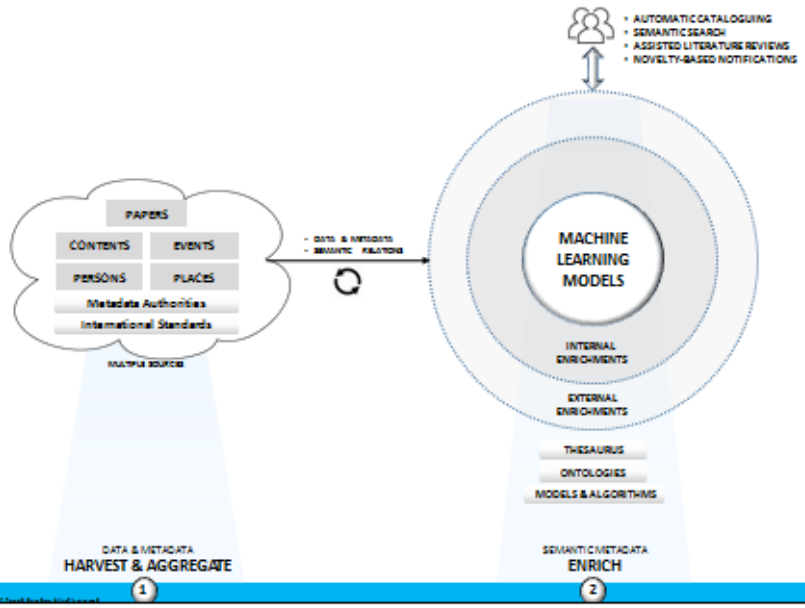
Research Goals:

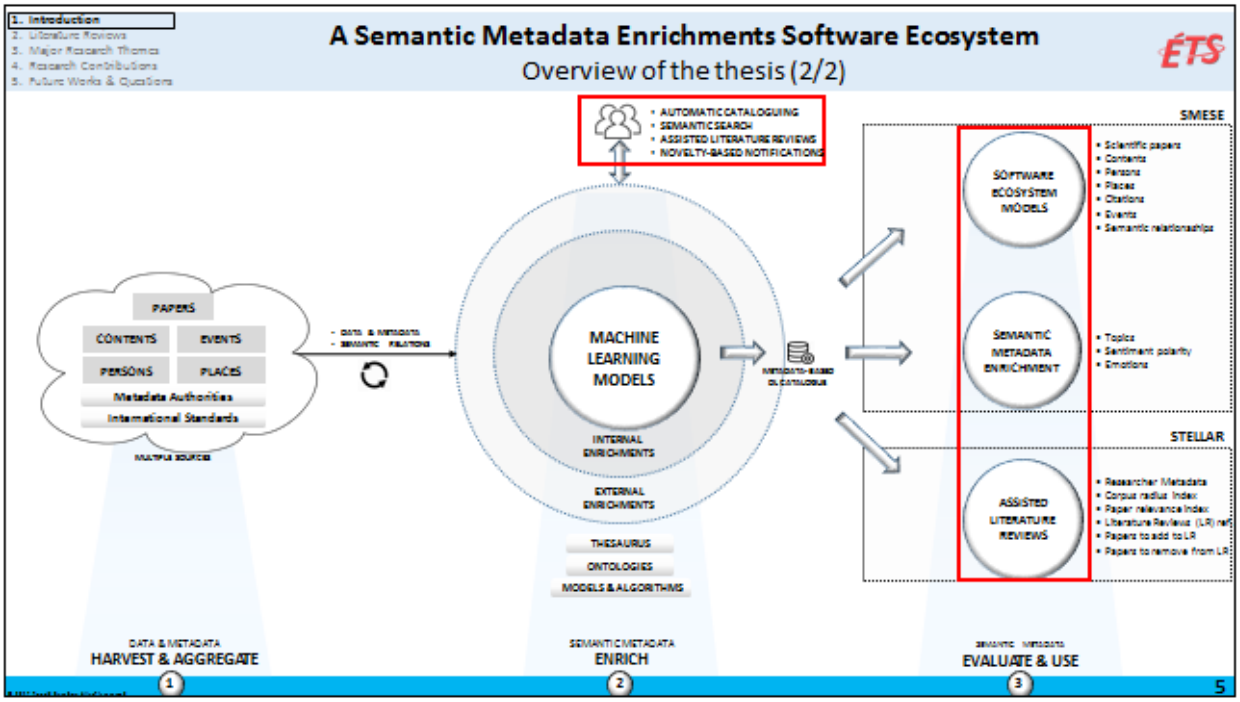
1. Proposal of a **unified metadata model** and mapping ontologies applied to DL.
2. Harvesting and ***semantic aggregation of metadata*** regardless of the sources.
3. **Semantic enrichments** of metadata by text analysis:
   1. ***hidden topics***,
   2. ***sentiment and emotions***.
4. Assist researchers in the evaluation of ***scientific papers relevancy, semantic similarity and ranking by topic*** or area of knowledge.

A Semantic Metadata Enrichments Software Ecosystem

Overview of the thesis (1/2)

ÉTS





A Semantic Metadata Enrichments Software Ecosystem

its Prototypes for Digital Libraries, Metadata Enrichments and Assisted Literature Reviews

ÉTS

1. Introduction

1. Context of the thesis (Motivations and Goals)
2. Overview of the thesis

2. Literature Reviews

1. Software Ecosystem Model
2. Semantic Metadata Enrichments
3. Assisted Literature Reviews

3. Major Research Themes

1. Software Ecosystem Model (SMESE V1)
2. Semantic Metadata Enrichments (SMESE V3)
3. Assisted Literature Reviews (STELLAR V1)

4. Research Contributions

1. Published articles related to this thesis
2. Software Ecosystem Models (SMESE V1)
3. Semantic Metadata Enrichments (SMESE V3)
4. Assisted Literature Reviews (STELLAR V1)

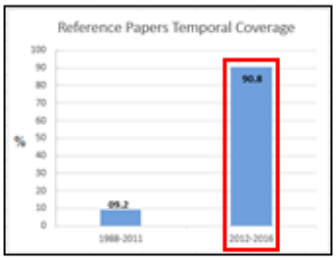
5. Future Works & Questions

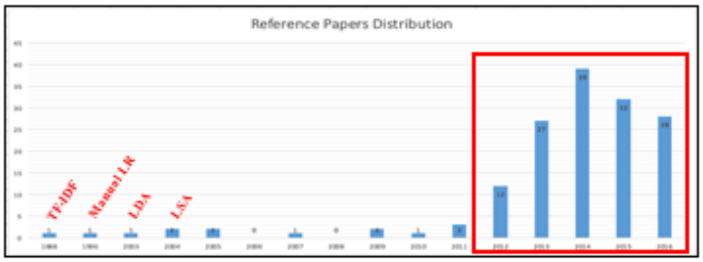
6

A Semantic Metadata Enrichments Software Ecosystem

Literature Review Papers Distribution

	#Papers	1988	1996	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016
Software Ecosystem Models	51											1	2	12	19	14	3
Semantic Metadata Enrichments	45	1		1	1	1				1			6	7	8	6	14
Assisted Literature Reviews	55		1		1	1		1		1	1	2	4	8	12	12	11
Relevant papers	151	1	1	1	2	2	0	1	0	2	1	3	12	27	39	32	28



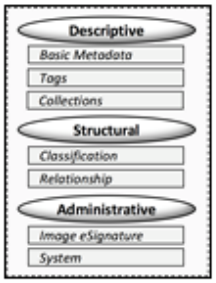


A Semantic Metadata Enrichments Software Ecosystem

1. Software Ecosystem Models: Related work limitations

Main drawbacks of SECO-based (Software Ecosystems) related to Digital Library (DL):

1. ***Do not*** offer a ***unified and interoperable DL metadata model***.
2. ***No*** architecture that simultaneously takes into account ***semantic metadata enrichments*** applied to many ecosystems.
3. ***No internal or external enrichments*** using a semantic model.
4. ***Do not*** propose ***a multi-domain ontology and thesaurus*** for semantic enrichment process.



Number	Model	Characteristics
1	SECO	Internal and external developers
2	SECO	Evaluative common technological platform
3	SECO	Controlled central part
4	SECO	Enable outside contributions and extensions
5	SECO	Variability-enabled architecture
6	SECO	Shared core assets
7	SECO	Automated and tool-supported product derivation
8	SECO	Outside contributions included in the main platform
9	SECO	Social network and IoT integration

SECO : Software Ecosystem
Lettner et al., 2014
Christian Doppler Laboratory MEVSS
Johannes Kepler University Linz, Austria

A Semantic Metadata Enrichments Software Ecosystem

2. Semantic Metadata Enrichments: Semantic Information Retrieval

ÉTS

Semantic information retrieval (SIR) models and their characteristics

- *Keywords*
- *Classifications*
- *Sentiments*
- *Emotions*
- *Concepts*

SIR Model	Keyword extraction	Classification	Sentiment analysis	Emotion analysis	Concept extraction
AlchemyAPI (http://www.alchemyapi.com/)	X	X	X	X	X
DBpedia Spotlight (https://github.com/dbpedia-spotlight)					X
Wikimeta (https://www.w3.org/2001/sw/wiki/Wikimeta)					X
Yahoo! Content Analysis API (https://developer.yahoo.com/contentanalysis/)		X			X
Open Calais (http://www.opencalais.com/)	X	X			X
Tone Analyzer (https://tone-analyzer-demo.mybluemix.net/)			X	X	
Zemanta (http://www.zemanta.com/)					X
Receptiviti (http://www.receptiviti.ai/)			X	X	
Apache Stanbol (https://stanbol.apache.org/)					X
Bitext (https://www.bitext.com/)			X		X
Mood patrol (https://market.mashape.com/soulhackerslabs/moodpatrol-emotion-detection-from-text)				X	
Aylien (http://aylien.com/)	X	X	X		
AIDA (http://senseable.mit.edu/aida/)					X
Wikifier (http://wikifier.org/)					X
TextRazor (https://www.textrazor.com/)					X
Synesketch (http://krcadinac.com/synesketch/)				X	
Toneapi (http://toneapi.com/)			X	X	

A Semantic Metadata Enrichment Software Ecosystem

2. Semantic Metadata Enrichments: Example of related works

ÉTS

1. Hidden topics detection

Works	Text size	Approaches	Semantic	Topic correlation	Machine Learning
(Dang et al., 2016)	short	Dynamic Bayesian networks	No	No	No
(Cigarrán et al., 2016)	short	Formal concept analysis (FCA)	No	No	No
(Sayyadi & Raschid, 2013)	long	Graph analysis methods	No	No	No
(Salatino & Motta, 2016)	long	Graph analysis methods	No	No	No
(P. Chen et al., 2016)	long	Probabilistic and graph analysis methods	No	No	No
(Hurtado et al., 2016)	long	Sentence-level association rule mining	No	No	No
(C. Zhang et al., 2016)	long	Probabilistic and graph analysis methods	No	No	No

2. Sentiment and emotion detection

Works	Text granularity	Approaches	Semantic	Valence	Emotion
(Cho et al., 2014)	Document	Keyword spotting		X	
(Bao et al., 2012)	Document	Statistical/Learning based methods	X		X
(Lei et al., 2014)	Phrase or clause	Lexical affinity			X
(Anusha & Sandhya, 2015)	Document	Statistical/Learning based methods	X		X
(Cambria et al., 2015)	Document	Statistical/Learning based methods	X		X

A Semantic Metadata Enrichment Software Ecosystem

2. Semantic Metadata Enrichments: Example of related works

ÉTS

1. Hidden topics detection

Works	Text size	Approaches	Semantic	Topic correlation	Machine Learning
(Dang et al., 2016)					
(Cigarrán et al., 2016)					
(Sayyadi & Raschid, 2013					
(Salatino & Motta, 2016)					
(P. Chen et al., 2016)					
(Hurtado et al., 2016)					
(C. Zhang et al., 2016)					

- ***Based on simple keyword extraction*** from text.
- ***Limited co-occurrence analysis***.
- Existing approaches focus mainly on ***detecting topics*** or ***frequent co-occurrence*** relations. ***Not focusing on latent modeling*** and ***large text.***
- Many works do not include machine learning to find new topics automatically.

2. Sentiment and emotion detection

Works
(Cho et al., 2014)
(Bao et al., 2012)
(Lei et al., 2014)
(Anusha & Sandhya,
(Cambria et al., 2015)

- Mainly use ***terms and frequency, pre-defined patterns*** and ***sentiment shifters*** *(+ or -)*. Most of the recent contributions are in terms of valence (positive or negative opinion).
- Do ***not combine sentiment and emotion analysis***.
- Do not take large text documents, they are ***sentence-based***.
- Do ***not allow human input***.

A Semantic Metadata Enrichment Software Ecosystem

3. Assisted Literature Reviews: Related work limitations

ÉTS

1. Many papers related to Literature Review ***don't use important related metadata***
   - Research domain
   - Research specific topic
   - Research title
   - Research description
   - Matching keywords
   - Notes from researchers

Approaches	Year of publication	Citation number	Reference	Venue type	Venue age	Authors' impact	Citation category	Venue impact	Authors' institutes	Citing document
PTRA (Hasson et al., 2014)	X	X		X						
ID3 (Rúbio & Gulo, 2016)	X	X	X	X						

PTRA: Hasson et al., 2014
College of Education of Pure Science, Basra University, Basra, Iraq

ID3: Rúbio and Gulo, 2016
Artificial Intelligence and Computer Science Laboratory
Faculty of Engineering, University of Porto, Porto, Portugal

2. Evaluation of scientific papers relevance ***do not take into account***:
   - Venue ***(publisher or conference) impact***
   - ***Authors affiliation and awards***
   - Distinction between ***co-authors*** and their order as co-authors
3. Specific ***structural organization of papers*** requires other method of text summarization.

A Semantic Metadata Enrichments Software Ecosystem

its Prototypes for Digital Libraries, Metadata Enrichments and Assisted Literature Reviews

ÉTS

1. Introduction

1. Context of the thesis (Motivations and Goals)
2. Overview of the thesis

2. Literature Reviews

1. Software Ecosystem Model
2. Semantic Metadata Enrichments
3. Assisted Literature Reviews

3. Major Research Themes

1. Software Ecosystem Model (SMESE V1)
2. Semantic Metadata Enrichments (SMESE V3)
3. Assisted Literature Reviews (STELLAR V1)

4. Research Contributions

1. Published articles related to this thesis
2. Software Ecosystem Models (SMESE V1)
3. Semantic Metadata Enrichments (SMESE V3)
4. Assisted Literature Reviews (STELLAR V1)

5. Future Works & Questions

13

1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions

A Semantic Metadata Enrichments Software Ecosystem

1. Software Ecosystem Model: master-catalogue contents classification

ÉTS

Unified Metadata model (DL):

1. Entity = 214
2. Metadata = 1,548
3. CrossWalk (Ontology) (using Protégé) = 26
4. International Library Standards = 3
5. Semantic relationship = 362
6. Based language = 3

EXTRACT of the METADATA MODEL of the SMESE's Master Catalogue

Calendars
- *Interests*
- *Library*
- *POI*
- *Rewards*
- *TV Channel*
- ...

Collections
- *Interests*
- *Library*
- *Organizations*
- *Personal*
- ...

Contents
- *Audio Books*
- *Books*
- *Cartographic Mat.*
- *Citations*
- *Comics*
- *Estampes*
- *Manga*
- *Microforms*
- *Movies (DVD)*
- *Music (CD)*
- *Musical Partitions*
- *Old Books*
- *Photos (Image)*
- *Press*
- *Serials*
- *Sounds*
- *Videos*
- ...

Documents
- *Google Doc*
- *Paint*
- *PDF*
- *Powerpoint*
- *Spreadsheet*
- *Word*
- ...

Events
- *Cinemas Rep.*
- *Expositions*
- *Libér Spirits*
- *News*
- *Notifications*
- *Press Conference*
- *Shows*
- *Spectacles*
- *Theaters*
- *TV Shows*
- ...

- ***ALRO***
- *Titre*
- *Topics*
- *Keywords*
- *#references*
- *Annotations*
- ...

Objects
- *Object*
- *Work of Art*
- ...

Persons
- *Actor*
- *Author*
- *Celebrity*
- *Musician*
- *Politician*
- *Producer*
- *Singer*
- *Students*
- *User*
- ...

Places
- *City*
- *Localization*
- *POI*
- ...

Products
- *Financial*
- *Groceries*
- *Hardware*
- *Natural Health*
- *Pharmacy*
- *Software*
- ...

Publications
- *Articles*
- *Education Programs*
- *Fact Sheets*
- *Questions/Answers*
- *Manuals*
- *Monographs*
- *Newsletters*
- *PostCards*
- *Posters*
- *Proceedings*
- *Thesis*
- ...

Rewards
- *Litterature*
- *Movies*
- *Music*
- *Nobel*
- ...

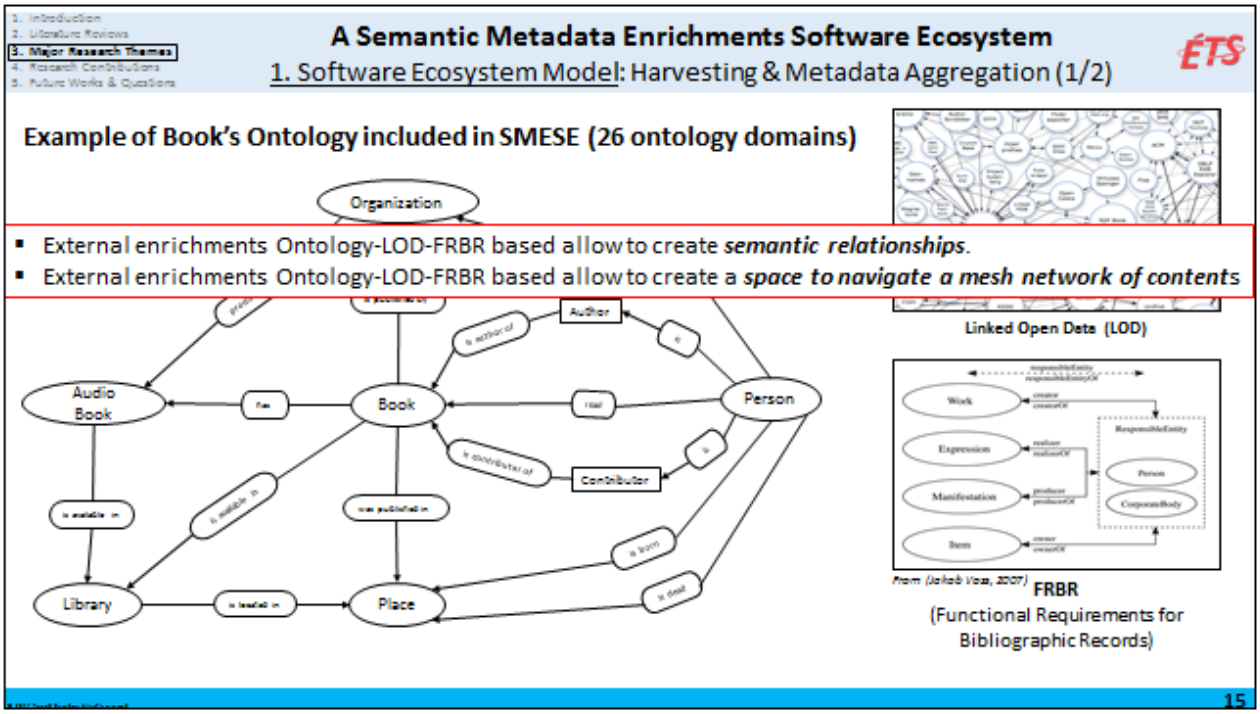
Resources
- *Online*
- *Physical*
- ...

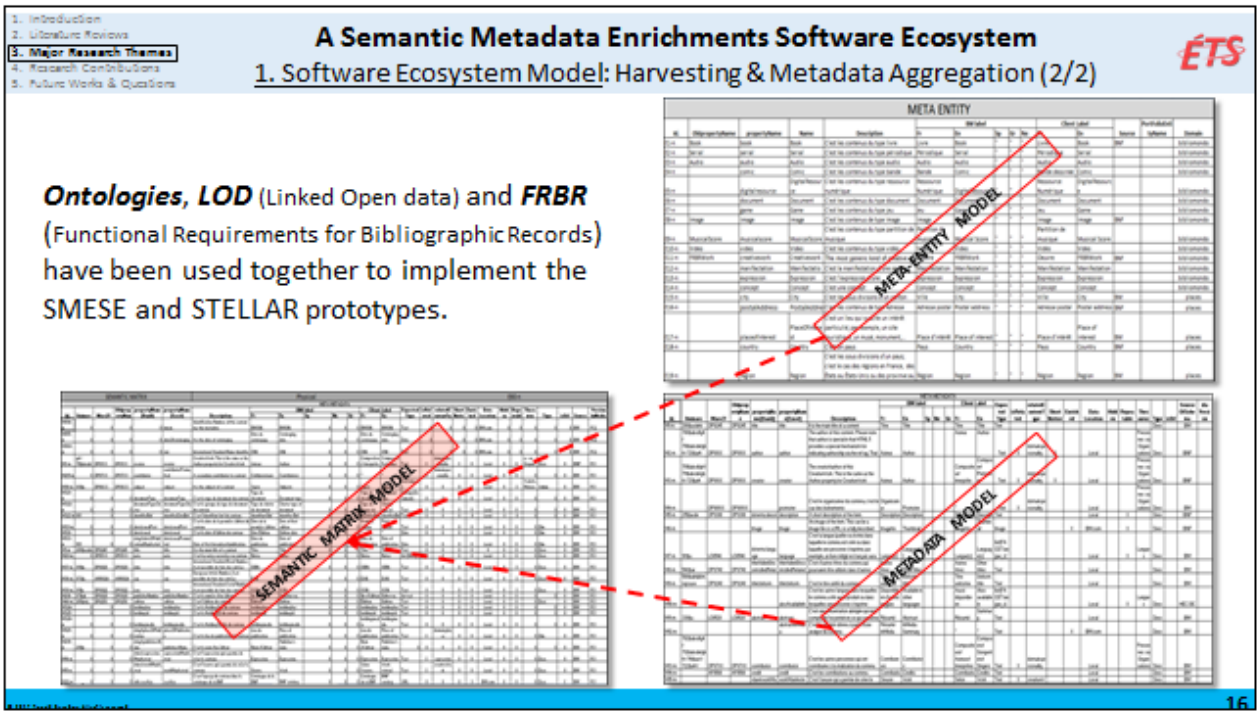
Subjects
- *Genomes*
- *MindMaps*
- *Ontologies*
- ...

Websites
- *Homework Help*
- *Youth*
- ...

Works
- *Concepts*
- *Expressions*
- *Manifestations*
- ...

13


1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichments Software Ecosystem
1. Software Ecosystem Model: Harvesting & Metadata Aggregation (1/2)
ÉTS
Example of Book's Ontology included in SMESE (26 ontology domains)
Organization
External enrichments Ontology-LOD-FRBR based allow to create semantic relationships.
External enrichments Ontology-LOD-FRBR based allow to create a space to navigate a mesh network of contents
Author
Audio Book
Book
Person
Contributor
Library
Place
Linked Open Data (LOD)
Work
Expression
Manifestation
Item
ResponsibleEntity
Person
CorporateBody
From (Jakob Voss, 2007)
FRBR
(Functional Requirements for Bibliographic Records)
15


1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichments Software Ecosystem
1. Software Ecosystem Model: Harvesting & Metadata Aggregation (2/2)
ÉTS
Ontologies, LOD (Linked Open data) and FRBR (Functional Requirements for Bibliographic Records) have been used together to implement the SMESE and STELLAR prototypes.
META ENTITY
META ENTITY MODEL
SEMANTIC MATRIX MODEL
METADATA MODEL
16

A Semantic Metadata Enrichments Software Ecosystem

2. Semantic Metadata Enrichments

1. ***Extension of the topic modeling with semantic information*** using words co-occurrence relations.
2. ***Definition of the latent co-occurrence relations*** between two terms are measured from an isolated term-term perspective.
3. Use of MLMs, semantic relations and sentiment lexicon to ***detect topics*** and ***sentiments in large documents***.

No	URL Sources	Status	%	Total Content	Total harvested
1	https://www.academia.edu/	h	1	15 363 735	142 217
2	https://www.researchgate.net/	h	0	100 000 000	27 984
3	http://www.opendoar.org/	h	0	235 828 824	385 155
4	http://www.amazon.com/	h	1	4 703 063	44 322
5	http://www.paulines.qc.ca/	h	76	171 120	130 000
6	http://exedre.leslibrairies.ca/	h	63	171 120	107 680
7	http://www.fnac.com/	h	67	156 224	104 109
8	http://www.librairiepoirier.ca/	f	100	176 162	176 162
9	http://www.archambault.ca/	f	100	165 405	165 405
10	http://www.renaudbray.com/accueil.aspx	f	100	347 380	347 380
11	http://www.usherbrooke.coop/	f	100	47 412	47 412
12	http://www.biblairie.qc.ca/	f	100	213	213
13	http://librairiemedieupaul.ca	f	100	29 938	29 938
14	https://www.leslibraires.fr/	f	100	888 750	888 750
15	http://www.mollat.com/	f	100	505 729	505 729
TOTAL				465 379 768	2 967 439

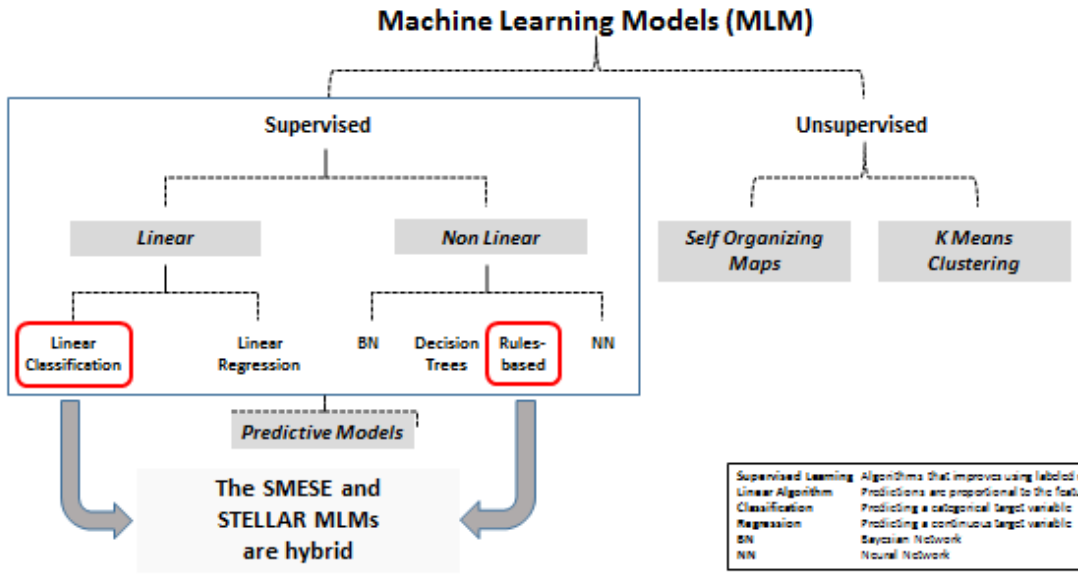
Status : f: finished and h: harvesting

Harvested documents (as march 2017)

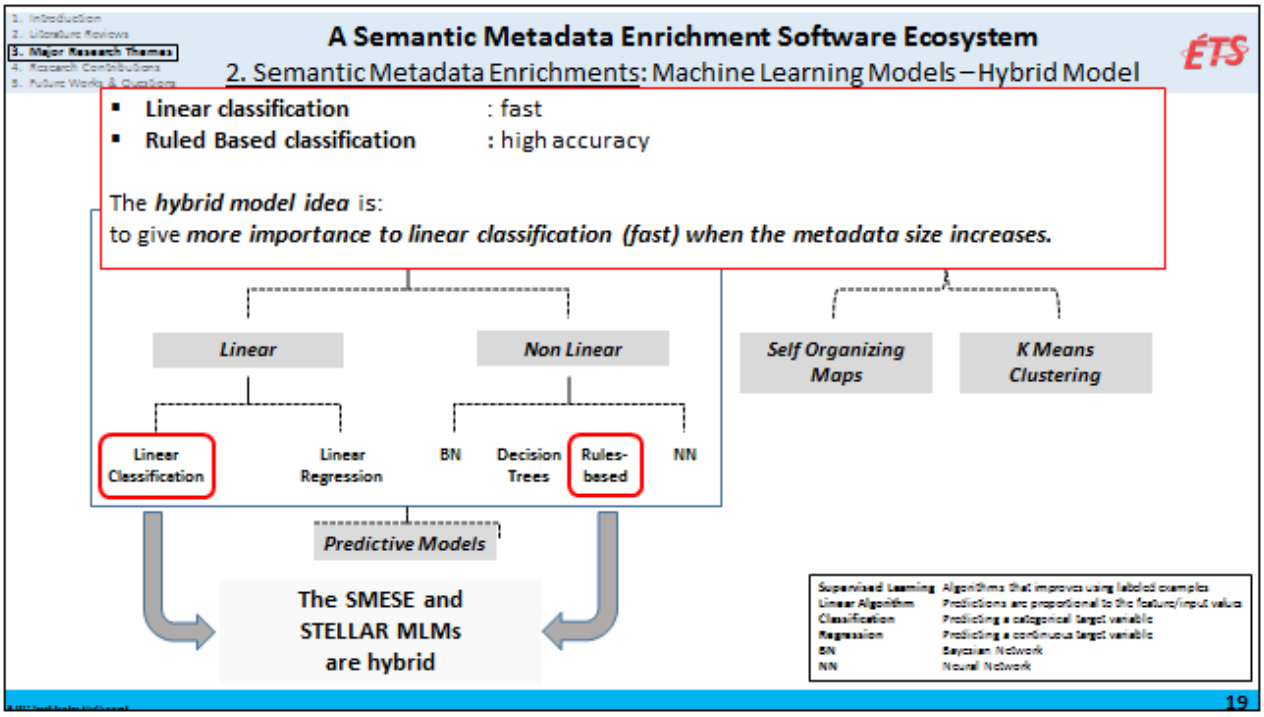
5 127 591
Harvested documents (as may 2017)

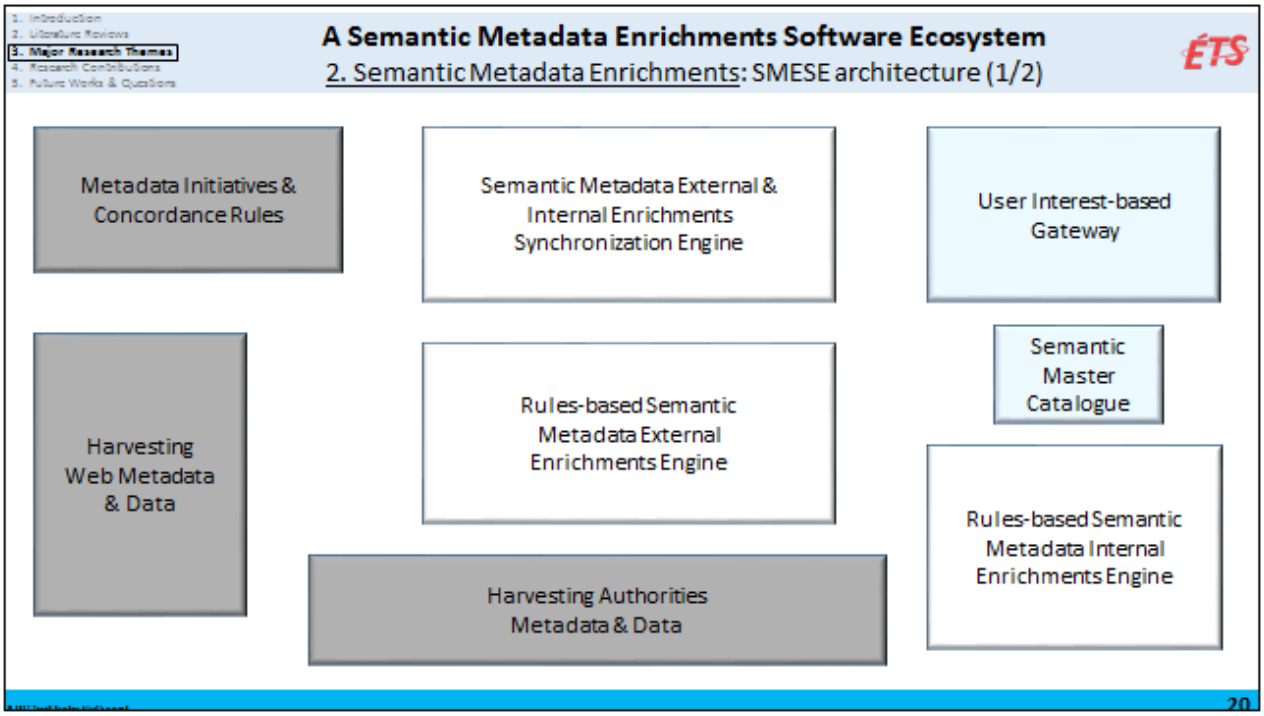
A Semantic Metadata Enrichment Software Ecosystem

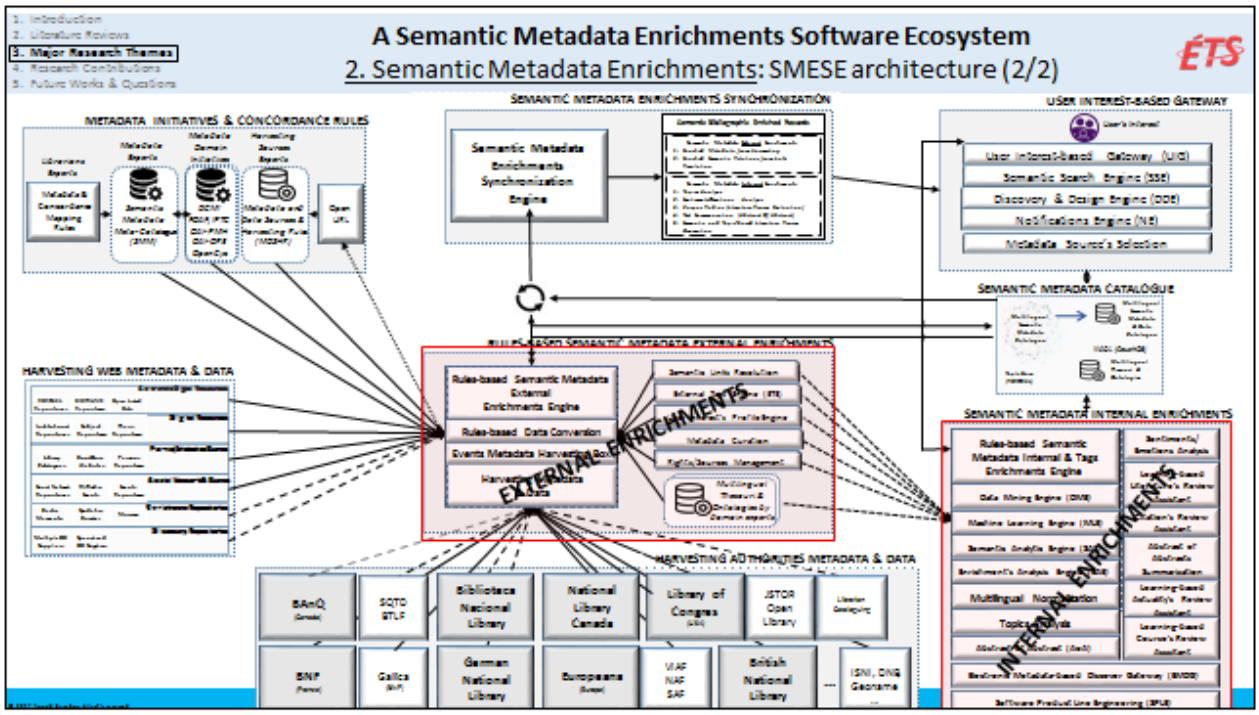
2. Semantic Metadata Enrichments: Machine Learning Models – Hybrid Model

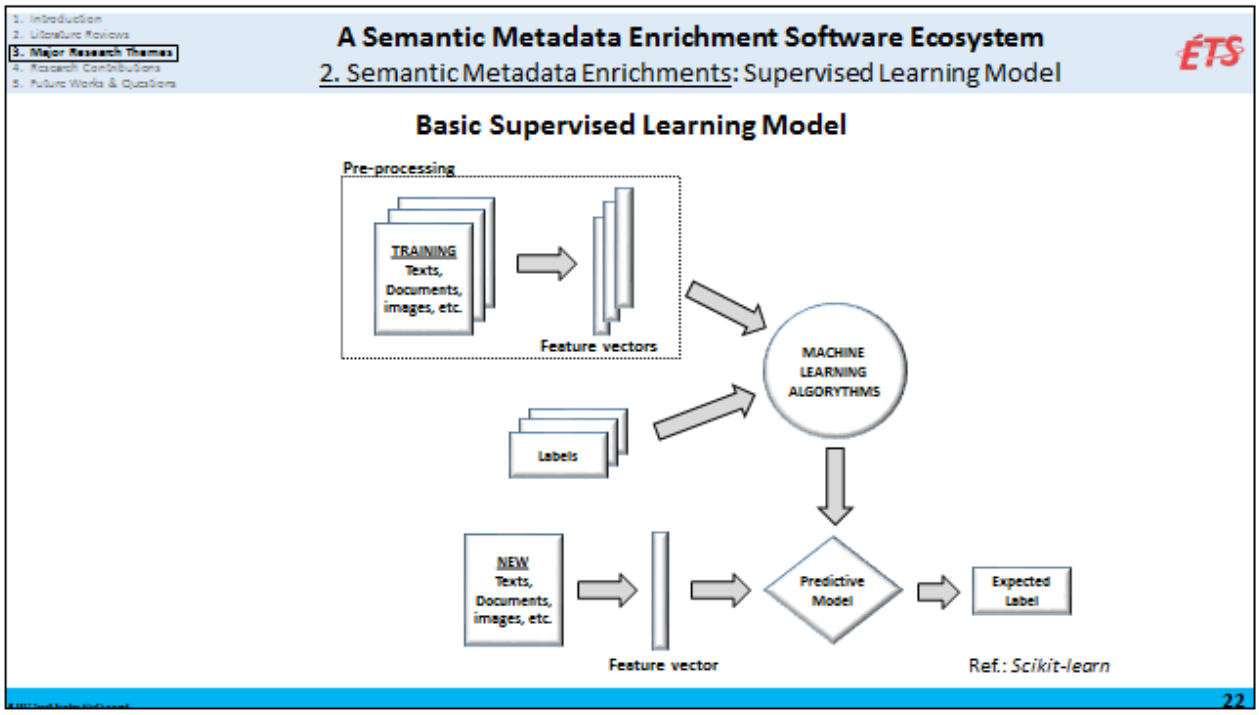


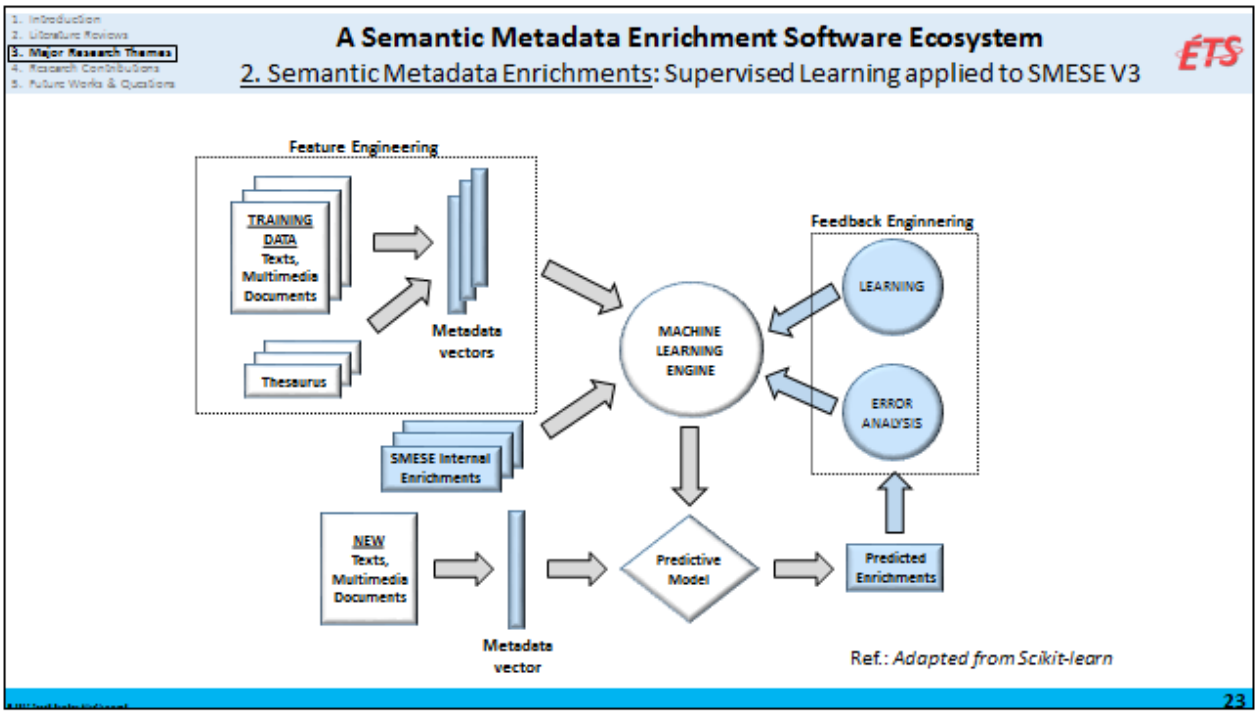
Supervised Learning	Algorithms that improves using labeled examples
Linear Algorithm	Predictions are proportional to the feature/input values
Classification	Predicting a categorical target variable
Regression	Predicting a continuous target variable
BN	Bayesian Network
NN	Neural Network

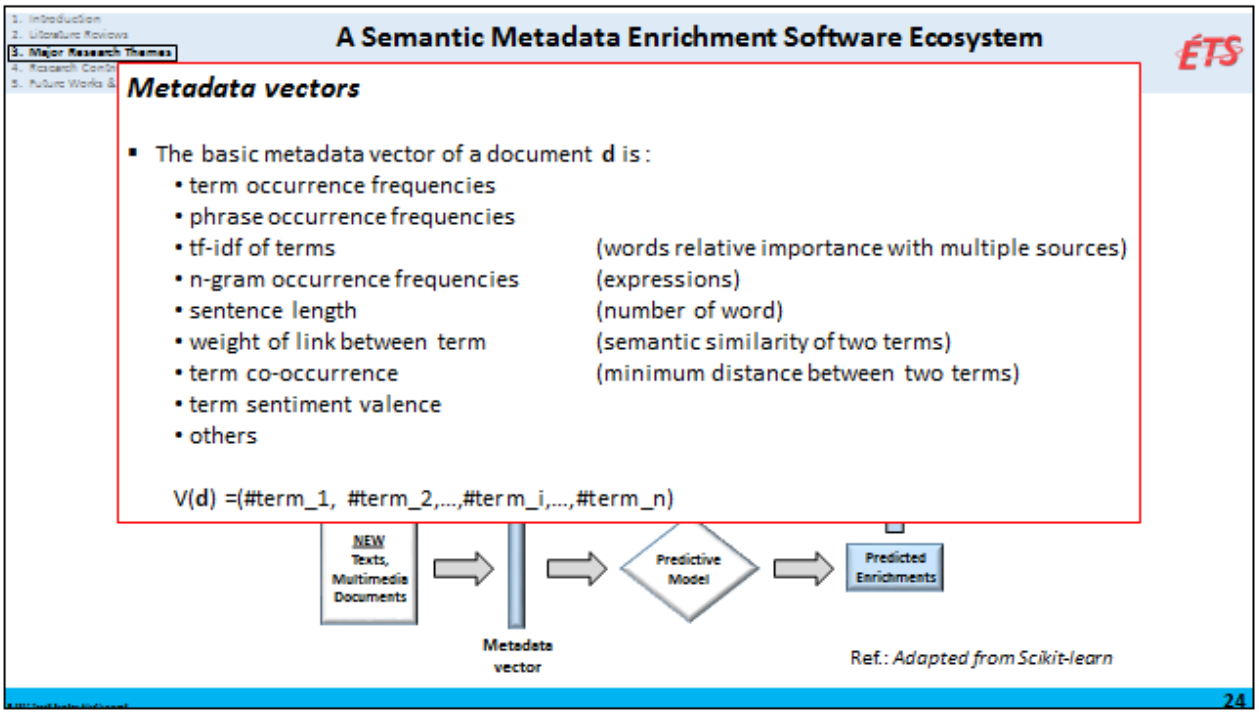

1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichment Software Ecosystem
2. Semantic Metadata Enrichments: Machine Learning Models – Hybrid Model
ÉTS
Linear classification : fast
Ruled Based classification : high accuracy
The hybrid model idea is:
to give more importance to linear classification (fast) when the metadata size increases.
Linear
Non Linear
Self Organizing Maps
K Means Clustering
Linear Classification
Linear Regression
BN
Decision Trees
Rules-based
NN
Predictive Models
The SMESE and STELLAR MLMs are hybrid
Supervised Learning Algorithms that improve using labeled examples
Linear Algorithm Predictions are proportional to the feature/input value
Classification Predicting a categorical target variable
Regression Predicting a continuous target variable
BN Bayesian Network
NN Neural Network
19

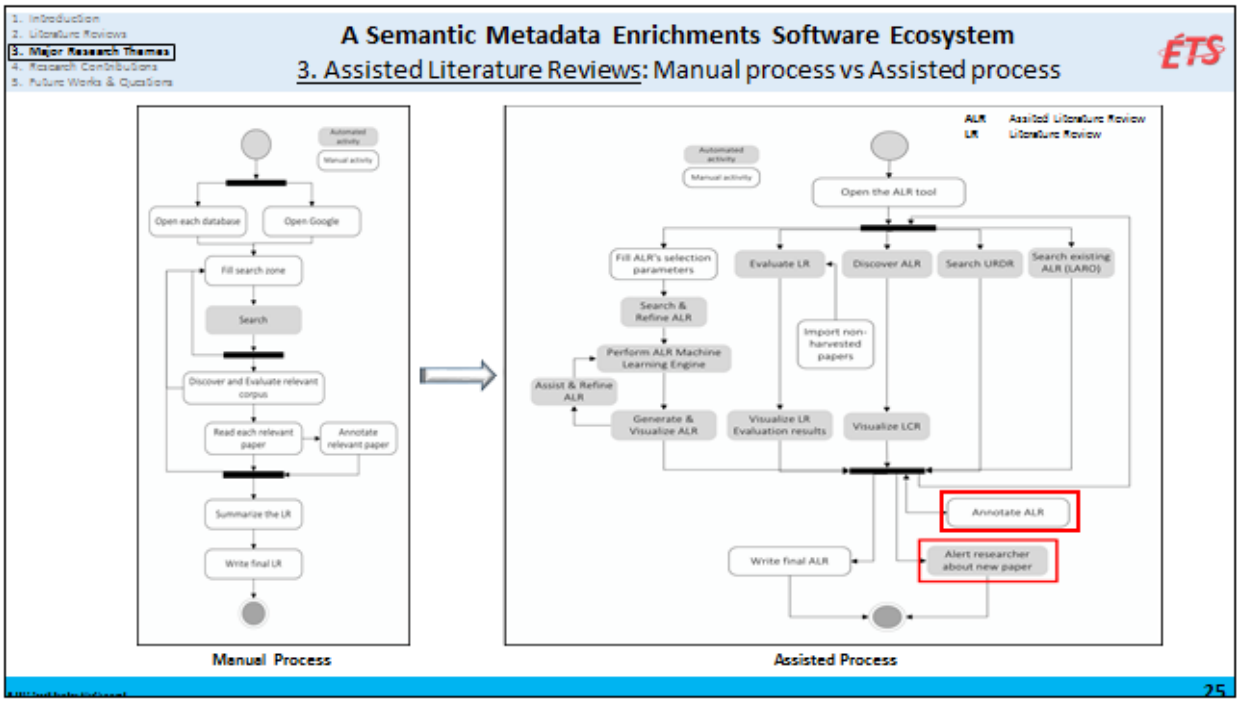

1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichments Software Ecosystem
2. Semantic Metadata Enrichments: SMESE architecture (1/2)
ÉTS
Metadata Initiatives & Concordance Rules
Semantic Metadata External & Internal Enrichments Synchronization Engine
User Interest-based Gateway
Semantic Master Catalogue
Harvesting Web Metadata & Data
Rules-based Semantic Metadata External Enrichments Engine
Rules-based Semantic Metadata Internal Enrichments Engine
Harvesting Authorities Metadata & Data
20

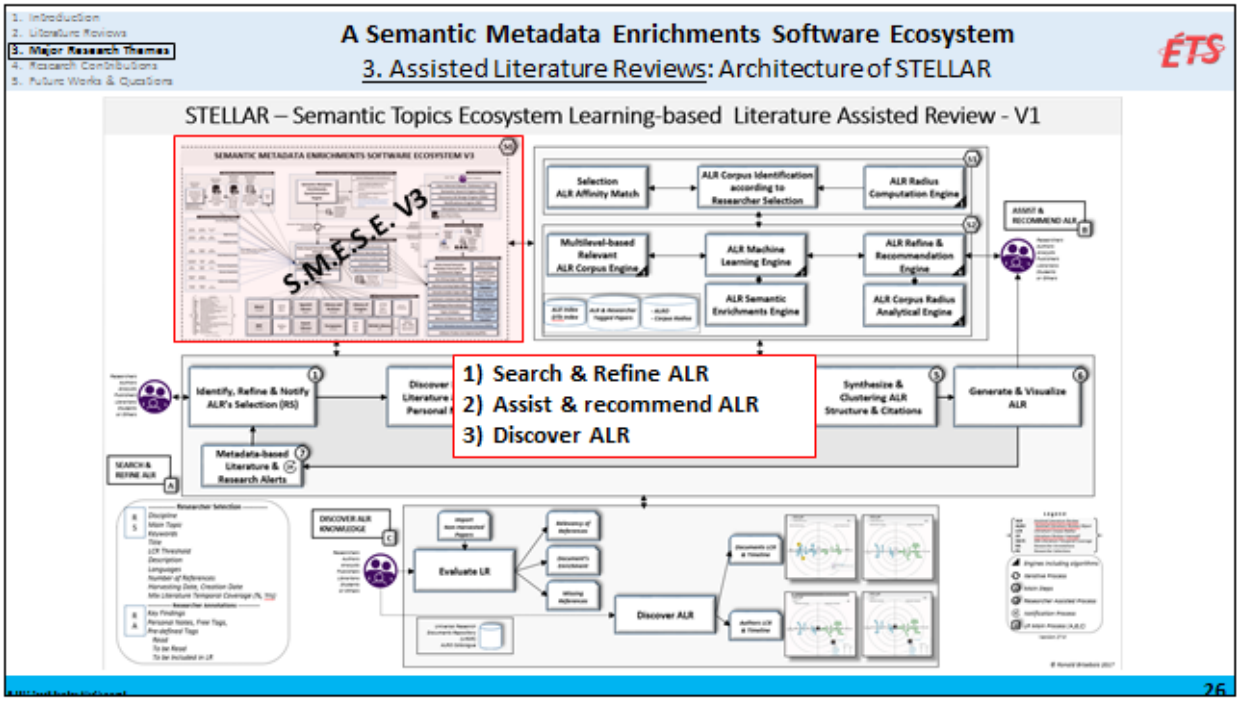

1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichments Software Ecosystem
2. Semantic Metadata Enrichments: SMESE architecture (2/2)
ÉTS
METADATA INITIATIVES & CONCORDANCE RULES
SEMANTIC METADATA ENRICHMENTS SYNCHRONIZATION
Semantic Metadata Enrichments Synchronization Engine
USER INTEREST-BASED GATEWAY
User Interest-Based Gateway (UIG)
Semantic Search Engine (SSE)
Discovery & Design Engine (DDE)
Notifications Engine (NE)
Metadata Source's Selection
SEMANTIC METADATA CATALOGUE
HARVESTING WEB METADATA & DATA
Rules-based Semantic Metadata External Enrichments Engine
Rules-based Data Conversion
EXTERNAL ENRICHMENTS
SEMANTIC METADATA INTERNAL ENRICHMENTS
Rules-based Semantic Metadata Internal & Tags Enrichments Engine
Data Mining Engine (DME)
Machine Learning Engine (MLE)
Multilingual Normalization
Topics Analysis
INTERNAL ENRICHMENTS
HARVESTING AUTHORITIES METADATA & DATA
BAnQ
SQTO BTLF
Biblioteca Nacional Library
National Library Canada
Library of Congress
JSTOR Open Library
BNF
Gallica
German National Library
Europeana
VIAF NAF SAF
British National Library
ISNI, DNB Geonams


1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichment Software Ecosystem
2. Semantic Metadata Enrichments: Supervised Learning Model
ÉTS
Basic Supervised Learning Model
Pre-processing
TRAINING Texts, Documents, images, etc.
Feature vectors
MACHINE LEARNING ALGORYTHMS
Labels
NEW Texts, Documents, images, etc.
Predictive Model
Expected Label
Feature vector
Ref.: Scikit-learn
22


1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichment Software Ecosystem
2. Semantic Metadata Enrichments: Supervised Learning applied to SMESE V3
ÉTS
Feature Engineering
TRAINING DATA Texts, Multimedia Documents
Thesaurus
Metadata vectors
Feedback Enginnering
LEARNING
ERROR ANALYSIS
MACHINE LEARNING ENGINE
SMESE Internal Enrichments
NEW Texts, Multimedia Documents
Metadata vector
Predictive Model
Predicted Enrichments
Ref.: Adapted from Scikit-learn
23


1. Introduction
2. Literature Reviews
3. Major Research Themes
A Semantic Metadata Enrichment Software Ecosystem
ÉTS
Metadata vectors
The basic metadata vector of a document d is :
• term occurrence frequencies
• phrase occurrence frequencies
• tf-idf of terms (words relative importance with multiple sources)
• n-gram occurrence frequencies (expressions)
• sentence length (number of word)
• weight of link between term (semantic similarity of two terms)
• term co-occurrence (minimum distance between two terms)
• term sentiment valence
• others
V(d) =(#term_1, #term_2,...,#term_i,...,#term_n)
NEW Texts, Multimedia Documents
Metadata vector
Predictive Model
Predicted Enrichments
Ref.: Adapted from Scikit-learn
24


1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichments Software Ecosystem
3. Assisted Literature Reviews: Manual process vs Assisted process
ÉTS
Open each database
Open Google
Fill search zone
Search
Discover and Evaluate relevant corpus
Read each relevant paper
Annotate relevant paper
Summarize the LR
Write final LR
Manual Process
ALR Assisted Literature Review
LR Literature Review
Open the ALR tool
Fill ALR's selection parameters
Evaluate LR
Discover ALR
Search URDR
Search existing ALR (LARO)
Search & Refine ALR
Import non-harvested papers
Perform ALR Machine Learning Engine
Assist & Refine ALR
Generate & Visualize ALR
Visualize LR Evaluation results
Visualize LCR
Annotate ALR
Write final ALR
Alert researcher about new paper
Assisted Process
25


1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichments Software Ecosystem
3. Assisted Literature Reviews: Architecture of STELLAR
ÉTS
STELLAR – Semantic Topics Ecosystem Learning-based Literature Assisted Review - V1
S.M.E.S.E. V3
Selection ALR Affinity Match
ALR Corpus Identification according to Researcher Selection
ALR Radius Computation Engine
Multilevel-based Relevant ALR Corpus Engine
ALR Machine Learning Engine
ALR Refine & Recommendation Engine
ALR Semantic Enrichments Engine
ALR Corpus Radius Analytical Engine
Identify, Refine & Notify ALR's Selection (RS)
Synthesize & Clustering ALR Structure & Citations
Generate & Visualize ALR
Metadata-based Literature & Research Alerts
1) Search & Refine ALR
2) Assist & recommend ALR
3) Discover ALR
Evaluate LR
Discover ALR
26

A Semantic Metadata Enrichments Software Ecosystem

3. Assisted Literature Reviews: Papers relevant to ALR

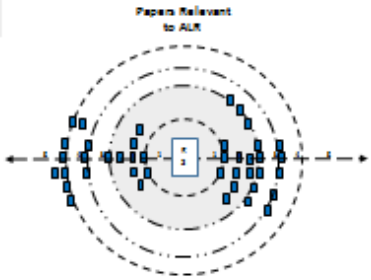
ÉTS

---------- Researcher Selection ----------

Discipline
Main Topic
Keywords
Title
Literature Corpus Radius (LCR) Threshold
Description
Languages
Number of References
Harvesting Date, Creation Date
Mix Literature Temporal Coverage (MLTC)

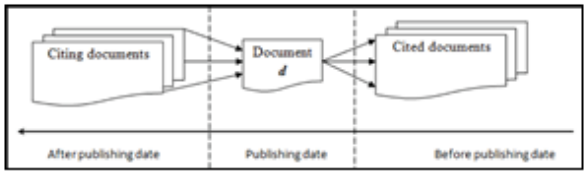
Unicity of STELLAR Model

LCR Threshold
Papers Relevant to ALR
Corpus Radius



Algorithms are based on:

- ***Researcher Selection*** (RS) and ***Researcher Annotations*** (RA)
- Library classification can enhance existing thesaurus model
- Machine Learning Models based on thesaurus and ontologies:
  a. Section recognition learning model
  b. Citation-based learning model:
  c. Text-based learning model:



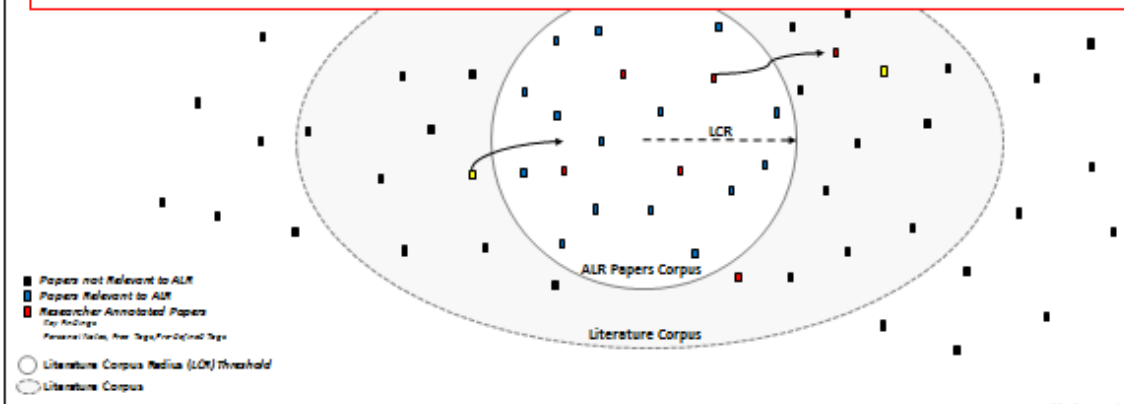
Paper Timeline Relationships

A Semantic Metadata Enrichments Software Ecosystem

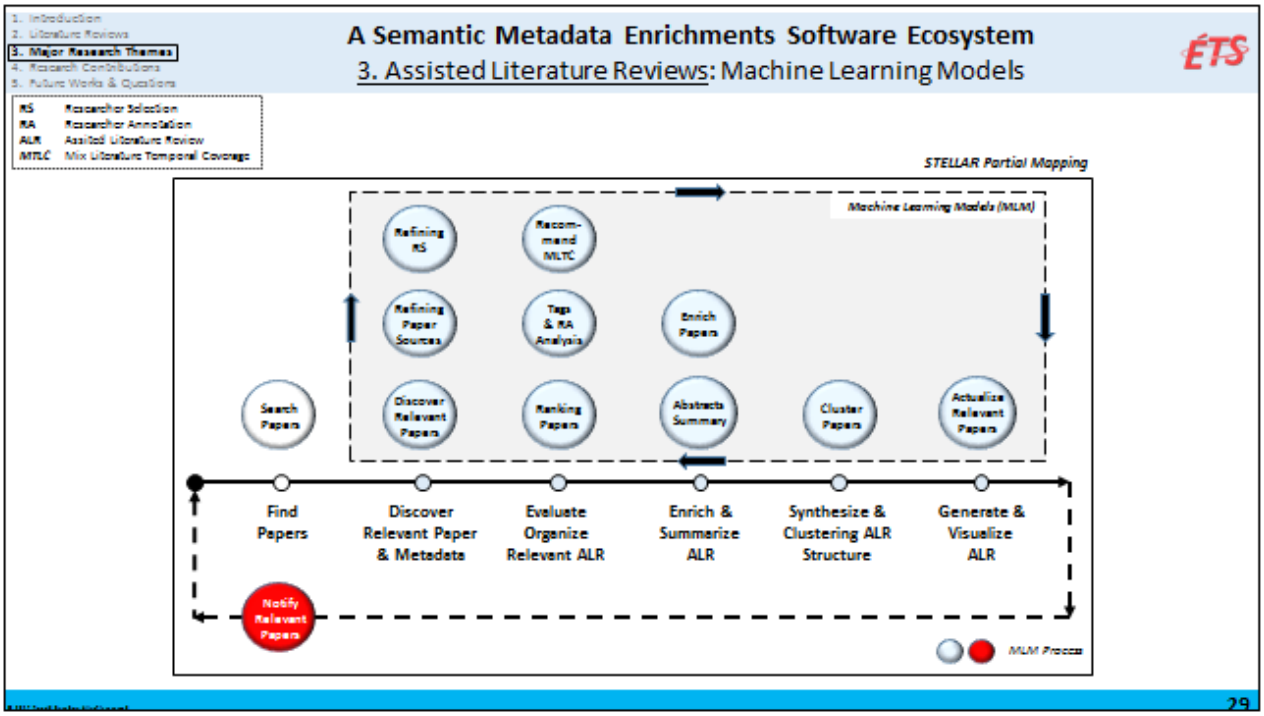
ÉTS

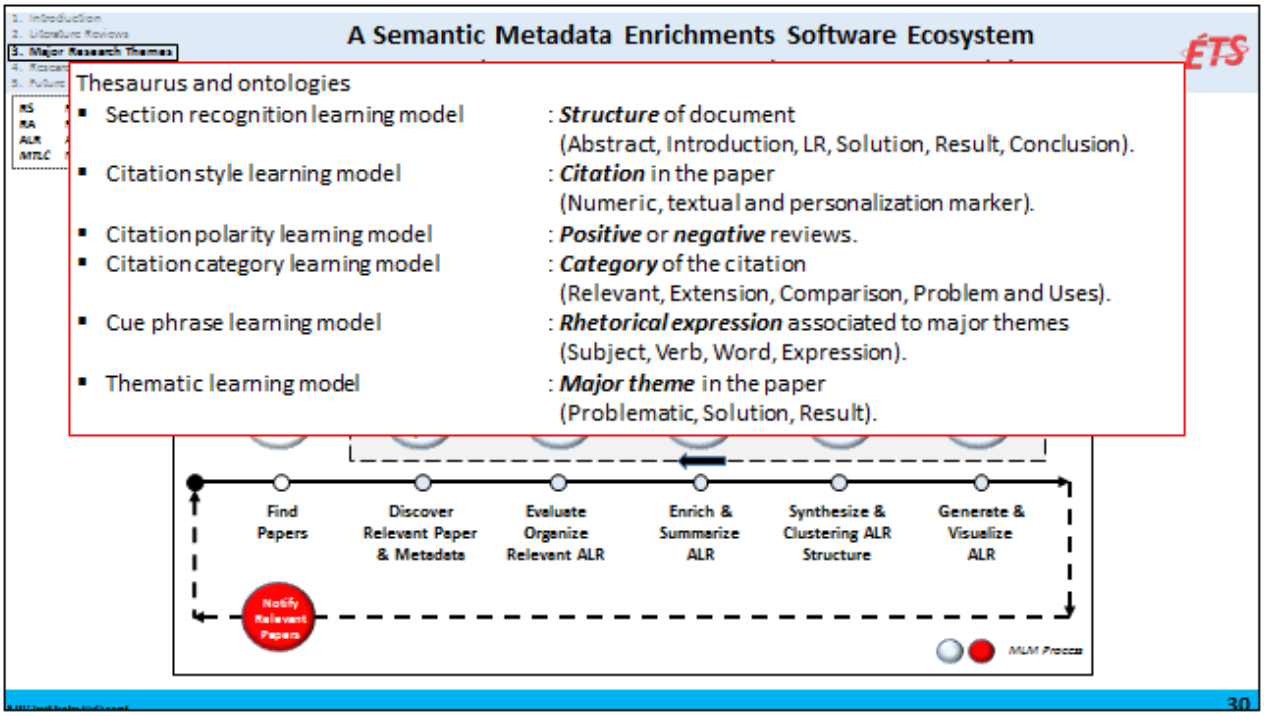
The goal is to limit the number of papers to those that are relevant and to rank them.

- ***Research documents Repository*** represents all papers regardless of their relevancy.
- The ***Literature Corpus*** contains all the papers regardless.
- The papers within ***Corpus Radius*** are those located in a circle with the specific corpus radius.
- ***Literature Corpus Radius*** measures the semantic relevancy of a paper according to the Researcher Selection.
- ***Researcher Annotations*** consist of researcher notes, tags and key findings.



Universal Research Documents Repository


1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
RS Researcher Selection
RA Researcher Annotation
ALR Assisted Literature Review
MTLC Mix Literature Temporal Coverage
A Semantic Metadata Enrichments Software Ecosystem
3. Assisted Literature Reviews: Machine Learning Models
ÉTS
STELLAR Partial Mapping
Machine Learning Models (MLM)
Refining RS
Recom-mend MLTC
Refining Paper Sources
Tags & RA Analysis
Enrich Papers
Search Papers
Discover Relevant Papers
Ranking Papers
Abstracts Summary
Cluster Papers
Actualize Relevant Papers
Find Papers
Discover Relevant Paper & Metadata
Evaluate Organize Relevant ALR
Enrich & Summarize ALR
Synthesize & Clustering ALR Structure
Generate & Visualize ALR
Notify Relevant Papers
MLM Process
29

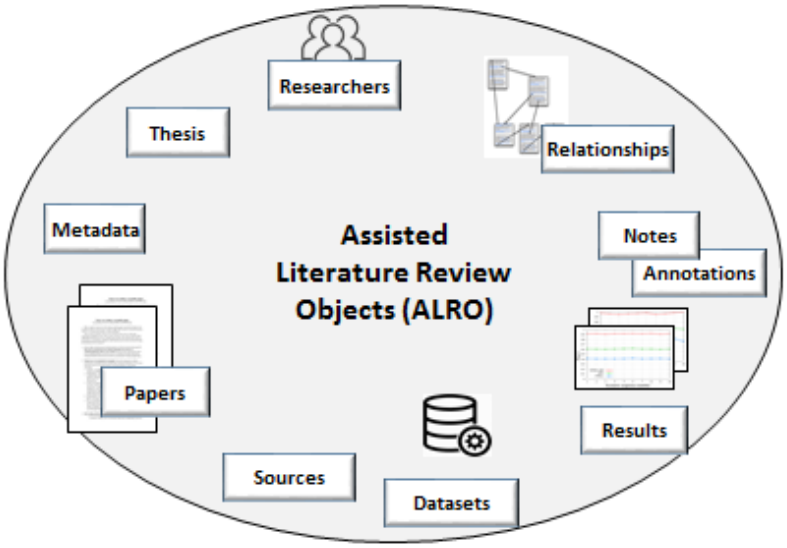

1. Introduction
2. Literature Reviews
3. Major Research Themes
A Semantic Metadata Enrichments Software Ecosystem
ÉTS
Thesaurus and ontologies
Section recognition learning model : Structure of document (Abstract, Introduction, LR, Solution, Result, Conclusion).
Citation style learning model : Citation in the paper (Numeric, textual and personalization marker).
Citation polarity learning model : Positive or negative reviews.
Citation category learning model : Category of the citation (Relevant, Extension, Comparison, Problem and Uses).
Cue phrase learning model : Rhetorical expression associated to major themes (Subject, Verb, Word, Expression).
Thematic learning model : Major theme in the paper (Problematic, Solution, Result).
Find Papers
Discover Relevant Paper & Metadata
Evaluate Organize Relevant ALR
Enrich & Summarize ALR
Synthesize & Clustering ALR Structure
Generate & Visualize ALR
Notify Relevant Papers
MLM Process
30

1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions

A Semantic Metadata Enrichments Software Ecosystem

3. Assisted Literature Reviews: ALRO

- ***Aggregation of ALR objects*** to form a reusable Assisted Literature Research Objects (ALRO).
- Catalogued and identified by a URI with ARK, so they can be ***shared, reused and cited***.
- Enable the verification of ***reproductibility*** of the results.



ARK Archive Resource Key
URI Unique Resource Identifier

Inspired from: http://www.researchobject.org/

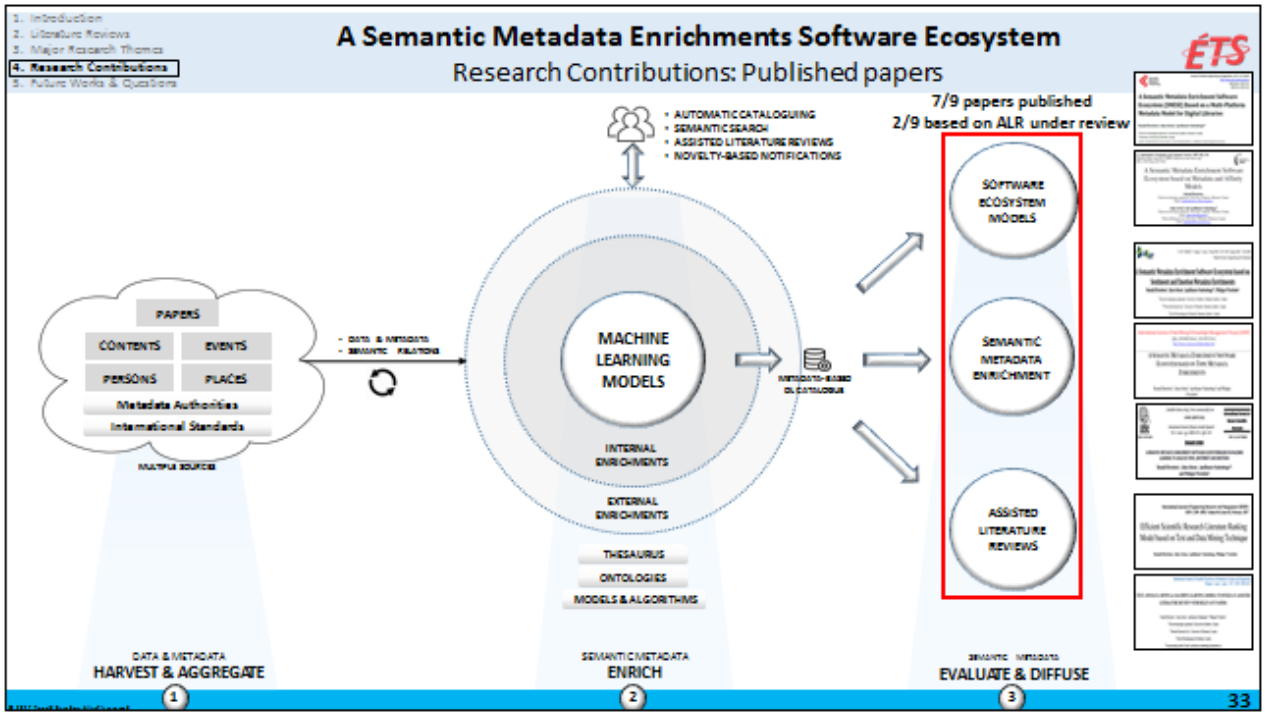
31

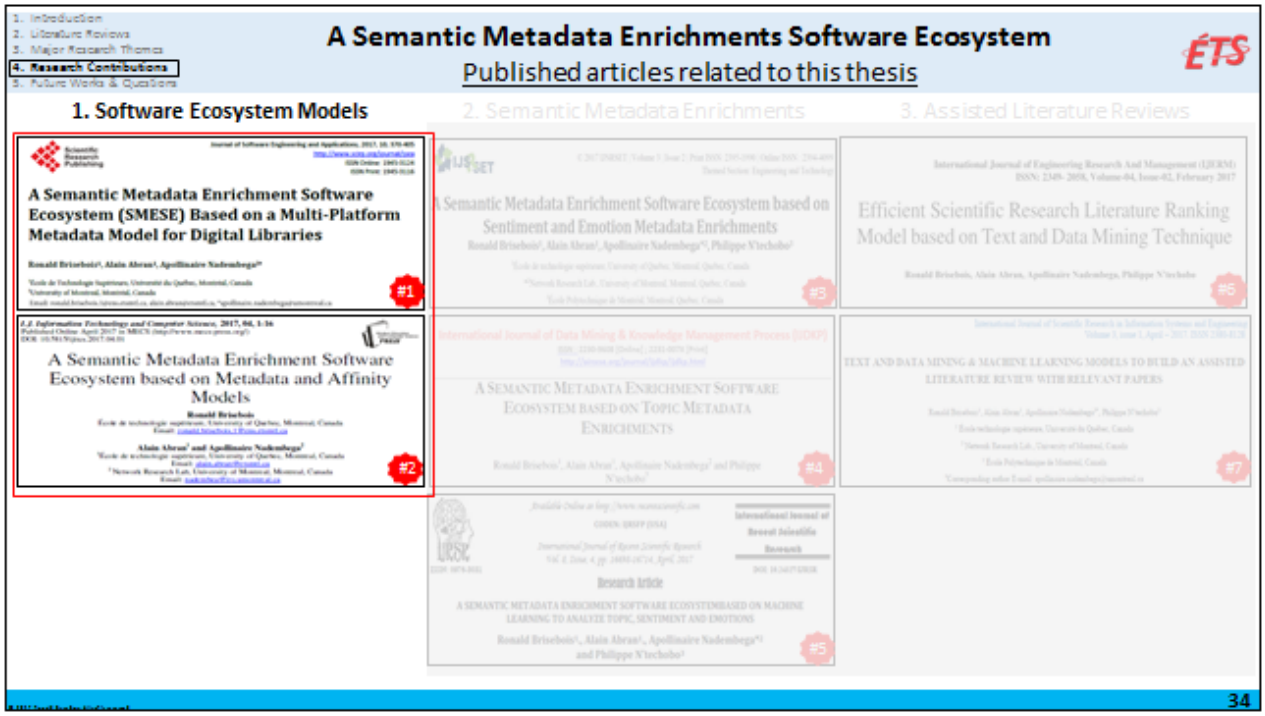
A Semantic Metadata Enrichments Software Ecosystem

its Prototypes for Digital Libraries, Metadata Enrichments and Assisted Literature Reviews

1. Introduction	1. Context of the thesis (Motivations and Goals) 2. Overview of the thesis	
2. Literature Reviews	1. Software Ecosystem Model 2. Semantic Metadata Enrichments 3. Assisted Literature Reviews	
3. Major Research Themes	1. Software Ecosystem Model 2. Semantic Metadata Enrichments 3. Assisted Literature Reviews	(SMESE V1) (SMESE V3) (STELLAR V1)
4. Research Contributions	1. Published articles related to this thesis 2. Software Ecosystem Models 3. Semantic Metadata Enrichments 4. Assisted Literature Reviews	 (SMESE V1) (SMESE V3) (STELLAR V1)
5. Future Works & Questions		

32


1. Introduction
2. Literature Reviews
3. Major Research Thrust
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichments Software Ecosystem
Research Contributions: Published papers
ÉTS
7/9 papers published
2/9 based on ALR under review
• AUTOMATIC CATALOGUING
• SEMANTIC SEARCH
• ASSISTED LITERATURE REVIEWS
• NOVELTY-BASED NOTIFICATIONS
PAPERS
CONTENTS
EVENTS
PERSONS
PLACES
Metadata Authorities
International Standards
MULTIPLE SOURCES
MACHINE LEARNING MODELS
INTERNAL ENRICHMENTS
EXTERNAL ENRICHMENTS
THESAURUS
ONTOLOGIES
MODELS & ALGORITHMS
SOFTWARE ECOSYSTEM MODELS
SEMANTIC METADATA ENRICHMENT
ASSISTED LITERATURE REVIEWS
DATA & METADATA
HARVEST & AGGREGATE
SEMANTIC METADATA
ENRICH
SEMANTIC METADATA
EVALUATE & DIFFUSE
1
2
3
33


1. Introduction
2. Literature Reviews
3. Major Research Thrust
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichments Software Ecosystem
Published articles related to this thesis
ÉTS
1. Software Ecosystem Models
2. Semantic Metadata Enrichments
3. Assisted Literature Reviews
A Semantic Metadata Enrichment Software Ecosystem (SMESE) Based on a Multi-Platform Metadata Model for Digital Libraries
#1
A Semantic Metadata Enrichment Software Ecosystem based on Metadata and Affinity Models
#2
A Semantic Metadata Enrichment Software Ecosystem based on Sentiment and Emotion Metadata Enrichments
#3
A SEMANTIC METADATA ENRICHMENT SOFTWARE ECOSYSTEM BASED ON TOPIC METADATA ENRICHMENTS
#4
A SEMANTIC METADATA ENRICHMENT SOFTWARE ECOSYSTEM BASEDON MACHINE LEARNING TO ANALYZE TOPIC, SENTIMENT AND EMOTIONS
#5
Efficient Scientific Research Literature Ranking Model based on Text and Data Mining Technique
#6
TEXT AND DATA MINING & MACHINE LEARNING MODELS TO BUILD AN ASSISTED LITERATURE REVIEW WITH RELEVANT PAPERS
#7
34

1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions

A Semantic Metadata Enrichments Software Ecosystem

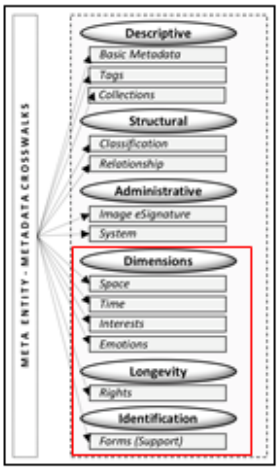
1. Software Ecosystem Model: SMESE V1

ÉTS

✓ *Goal = Proposal of a unified metadata model and mapping ontologies applied to DL*

Number	Model	Characteristics	
1	SECO	Internal and external developers	
2	SECO	Evaluative common technological platform	
3	SECO	Controlled central part	
4	SECO	Enable outside contributions and extensions	
5	SECO	Variability-enabled architecture	
6	SECO	Shared core assets	
7	SECO	Automated and tool-supported product derivation	
8	SECO	Outside contributions included in the main platform	
9	SECO	Social network and IoT integration	
10	SMESE	Semantic Metadata Internal Enrichments	X
11	SMESE	Semantic Metadata External Enrichments	X
12	SMESE	User Interest Affinity Model	X

Software Ecosystem Model (SECO)



Standard Metadata Description

The ***unified meta-model*** allows to build a ***Matrix of entity-metadata***:

Entity	= 214
Metadata	= 1,548
CrossWalk (Ontology)	= 26
International DL Standards	= 3
Semantic relationship (Meta)	= 362

1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions

A Semantic Metadata Enrichments Software Ecosystem

Published articles related to this thesis

ÉTS

1. Software Ecosystem Models | **2. Semantic Metadata Enrichments** | 3. Assisted Literature Reviews

1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions

A Semantic Metadata Enrichments Software Ecosystem

2. Semantic Metadata Enrichments: Paper selection

ÉTS

	Authors	Algorithms	Positive	Negative
1	David M. Blei et al., 2003	latent Dirichlet allocation (LDA)	• Texte plus long, version originale • Méthode prédominante • Modèle prédictive probabiliste	• +15 years
2	Bijalwan et al.,	k-nearest neighbors (K	• Récent (- 5 years)	• Temps d'obtention de résultats très grand (very high time complexity) • Juste une étude comparative
3	Dang et al., 2016	Bayesian model	• Recent (- 5 years)	• Texte court (micro-blogging) • Limit of 2 feature vectors
4	Zhang Chen et al. 2016	LDA-IG	• Recent (- 5 years), Texte plus long • Combine LDA et Graphes de mot clé • Extension of KeyGraph • Modèle prédictive probabiliste	-
5	Cigarrán et al., 2016	formal concept analysis	• Recent (- 5years)	• Texte court (tweet)
6	Cotelo et al., 2016	-	• Recent (- 5 years)	• Texte court (tweet)
7	Salatino & Motta, 2016	-	• Recent (- 5 years) • Texte plus long	• Extraction simple par mot-clé du texte. • N'utilise pas les relations sémantiques • Suppose que les mot clés sont des topic
8	Sayyadi & Raschid, 2013	KeyGraph	• Recent (- 5years), Texte plus long • Model basé sur les graphes de mot clé • Co-occurrence des mots clé • Similarité contextuelle, Analyse de graphe	• Extraction simple par mot-clé du texte
9	Chen et al., 2016	hierarchical latent tree models (HLTM)	• Récent (- 5 years), Texte plus long • Modelé basé sur les arbres hiérarchique des topics	-
10	Hurtado et al., 2016	Rule-based	• Récent (- 5 years) • Cooccurrence des mots clé • Utiliser l'information contextuelle	• Texte court (sentence-level) • Utilisation algorithme non conçu par les auteurs

1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions

A Semantic Metadata Enrichments Software Ecosystem

2. Semantic Metadata Enrichments: Paper selection

ÉTS

	Authors	Algorithms	Positive	Negative
1	David M. Blei et al., 2003	latent Dirichlet allocation (LDA)	• Texte plus long, version originale • Méthode prédominante • Modèle prédictive probabiliste	• +15 years
2	Bijalwan et al	k-nearest neighbors (K	• Récent (- 5 years)	• Temps d'obtention de résultats très grand
3	Dang et a			
4	Zhang Ch			
5	Cigarrán			
6	Cotelo et			
7	Salatino			
8	Sayyadi & Raschid, 2013	KeyGraph	• Recent (- 5years), Texte plus long • Model basé sur les graphes de mot clé • Co-occurrence des mots clé • Similarité contextuelle, Analyse de graphe	• Extraction simple par mot-clé du texte
9	Chen et al., 2016	hierarchical latent tree models (HLTM)	• Récent (- 5 years), Texte plus long • Modelé basé sur les arbres hiérarchique des topics	-
10	Hurtado et al., 2016	Rule-based	• Récent (- 5 years) • Cooccurrence des mots clé • Utiliser l'information contextuelle	• Texte court (sentence-level) • Utilisation algorithme non conçu par les auteurs

Topics Detection

Selection criteria:

Large Text

Recent except LDA who is an older algorithm; LDA is popular and has many publications.

Used best techniques (graph, tree, probability and hybrid):

LDA	is Graph-based and statistical
LDA-IG	is Hybrid (Graph-based and probabilistic based)
KeyGraph	is Graph-based
HLTM	is Tree-based

1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions

A Semantic Metadata Enrichments Software Ecosystem

2. Semantic Metadata Enrichments: Papers selection

ÉTS

	Authors	Algorithms	Positive	Negative
1	Cho et al., 2014	Keyword spotting	• Récent (- 5 years) • Dictionnaire (lexicons) de sentiment	• Limité au positive vs. Négative • Pas de valence sur les émotions
2	Bao et al., 2012	Learning based model emotion-topic model (ETM-LDA)	• Récent (- 5 years), Large Text • Modèle d'apprentissage (ML), emotions • Dictionnaire (lexicons) word-emotion • Combinaison de LDA et Bayesian model • Basé sur relation sémantique	• Pas de valence sur les émotions
3	Lei et al., 2014	lexicon-based	• Récent (- 5 years) • Lexicons of correspondance word-emotion	• No defintion of feature, 4 Emotions only • Analyse les expressions, No Valence
4	Anusha & Sandhya, 2015)	Learning based model Not name, so we call it Approach (AP)	• Récent (- 5 years), Large Text • Combine Machine learning (ML) and (NLP) • Basé sur relation sémantique • Sentiment Lexicons	• Pas de valence sur les émotions
5	Cambria et al., 2015	Learning based model TSVD	• Recent (- de 5 ans) • Fast	• Limited SVD for word-sentence, No Valence • Very limited Lexicons (AffectNet)
6	- L. Chen, Qi, & Wang, 2012; - Ghiassi et al., 2013; - Quan & Ren, 2014		• Récent (- de 5 ans) • Analyse les émotions au lieu de sentiment uniquement	• Analyse les mots au lieu des documents • Pas de valence sur les émotions
7	Tan, Na, Theng, & Chang,2012		• Récent (- de 5 ans), emotions	• Expressions analysis, No valence
8	- Abdul-Mageed, Diab, & Kübler, 2014; - Appel et al., 2016; Desmet & Hoste, 2013; - Niu et al., 2016; - Patel & Madia, 2016		• Récent (- de 5 ans) • Analyse les émotions au lieu de sentiment uniquement	• Analyse les phrases au lieu des documents • Pas de valence sur les émotions
9	Chen et al., 2016	hierarchical latent tree models (HLTM)	• Recent (- 5 years), Large text • Hierarchical Trees of topics	-

1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions

A Semantic Metadata Enrichments Software Ecosystem

2. Semantic Metadata Enrichments: Papers selection

ÉTS

	Authors	Algorithms	Positive	Negative
1	Cho et al., 2014	Keyword spotting	• Récent (- 5 years) • Dictionnaire (lexicons) de sentiment	• Limité au positive vs. Négative • Pas de valence sur les émotions
2	Bao et al., 2012	Learning based model emotion-topic model	• Récent (- 5 years), Large Text • Modèle d'apprentissage (ML), emotions	• Pas de valence sur les émotions
3	Lei et al., 20			y
4	Anusha & Sa			
5	Cambria et a			nce
6	- L. Chen, Qi - Ghiassi et a - Quan & Re			
7	Tan, Na, Theng, & Chang,2012		• Récent (- de 5 ans), emotions	• Expressions analysis, No valence
8	- Abdul-Mageed, Diab, & Kübler, 2014; - Appel et al., 2016; Desmet & Hoste, 2013; - Niu et al., 2016; - Patel & Madia, 2016		• Récent (- de 5 ans) • Analyse les émotions au lieu de sentiment uniquement	• Analyse les phrases au lieu des documents • Pas de valence sur les émotions
9	Chen et al., 2016	hierarchical latent tree models (HLTM)	• Recent (- 5 years), Large text • Hierarchical Trees of topics	-

Emotion and Sentiment

Selection criteria:

Recent
Large text
With the most used technics - Learning based
Used semantic relationships
Used an enriched lexicons of sentiments

ETM-LDA
AP

A Semantic Metadata Enrichments Software Ecosystem

2. Semantic Metadata Enrichments: SMESE V3 (1/3)

ÉTS

✓ *Goal = Proposal Semantic enrichments of metadata by text analysis: hidden topic, sentiment and emotions*

The contributions of SMESE V3 are:

1. Enhance the discovery of topic, sentiment and emotion metadata hidden within the text or linked to multimedia structure using the proposed algorithms:

Approach	Granularity	Description	Training phase	Refining	Semantic	Topic correlation	Domain knowledge
LDA-IG	Document	Probabilistic and graph based	Yes	No	No	No	No
KeyGraph	Document	Graph based	Yes	No	No	No	No
LDA	Document	Probabilistic based	No	No	No	No	No
HLTM	Document	Probabilistic and graph based	Yes	No	No	No	No
SATD	Configurable as desired	Semantic, probabilistic and graph based	Yes	Yes	Yes	Yes	Yes

BM-SATD - ***Scalable Annotation-based Topic Detection***

LDA-IG: Zhang Chen et al. 2016
Institute of Software, Chinese Academy of Science Beijing, China
KeyGraph: HASSAN SAYYADI and LOUIQA RASCHID, 2013. Un. Maryland
LDA: David M. Blei et al., 2003, University of California, Berkeley, CA, USA
HLTM: Peixian Chen et al., 2016, The Hong Kong University, Hong Kong

Approach	Granularity	Approach	Training phase	Refining	The saurus	Topic modeling	Emotions number
AP	Document	Learning based	Yes	No	5	No	8
ETM-LDA	Document	Keyword based	Yes	No	6	Yes	8
SSEA	Configurable as desired	Keyword and rule based	Yes	Yes	1, 2, 3, and 4	Yes	8

1-WordNet 2-WordNet-Affect 3-SentiWordNet;
4-NRC Emotion Lexicon 5- Stanford CoreNLP |6-Gibbs sampling.

BM-SSEA - **Semantic Sentiment and Emotion Analysis**

AP: Vajrapu Anusha and Banda Sandhya, 2015
Maturi Venkata Subba Rao Engineering College
Engineering college,, Hyderabad, India
ETM-LDA: Shenghua Bao et al. 2012
IBM Research-China.

A Semantic Metadata Enrichments Software Ecosystem

2. Semantic Metadata Enrichments: BM-SATD and BM-SSEA datasets (2/3)

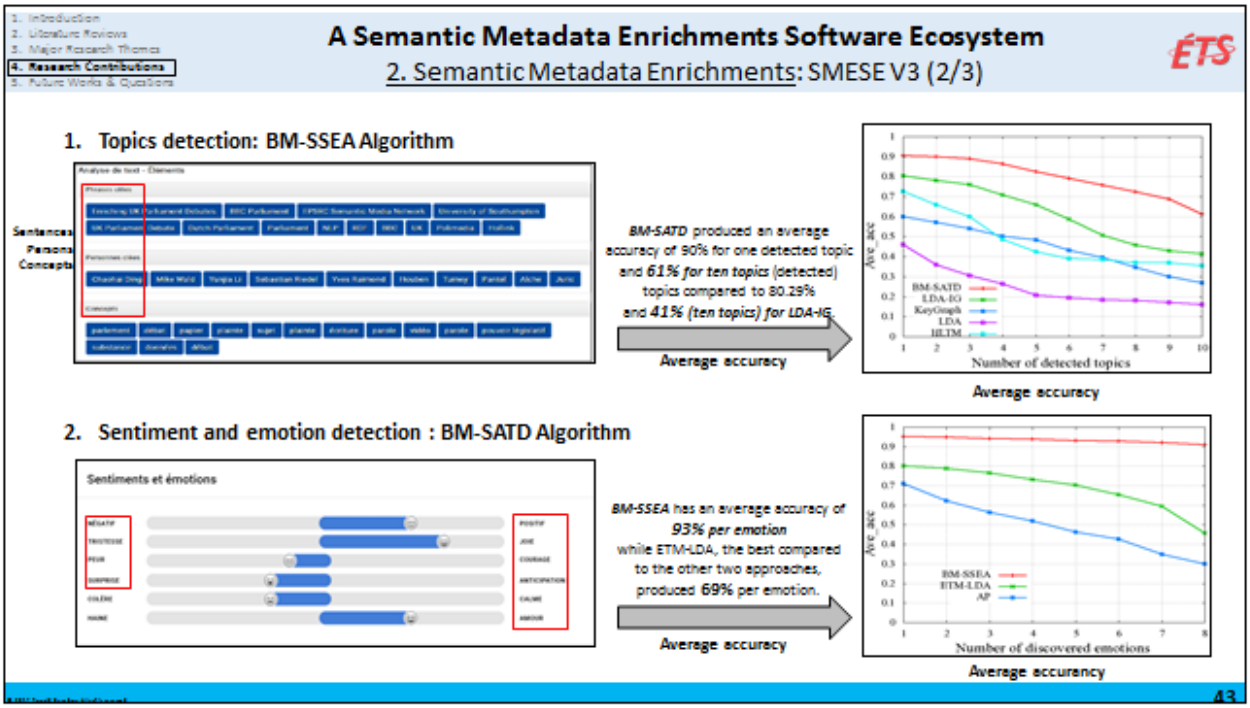
ÉTS

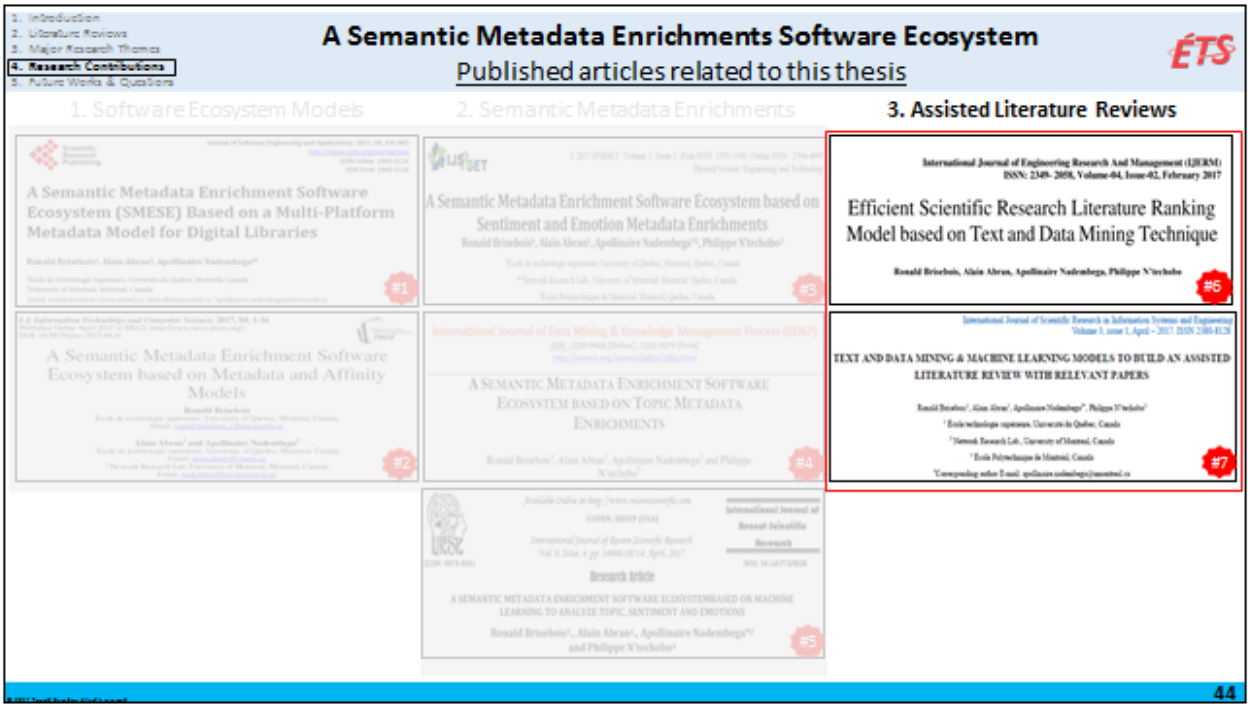
The contributions of SMESE V3 are:

1. Implementation of these prototypes for semantic metadata internal enrichment including algorithms **BM-SATD** and **BM-SSEA**.
2. Dataset used for simulation and prototypes of **BM-SATD** and **BM-SSEA**

Documents number (25,000)

- ***Training documents*** number : 15,000
- ***Test documents*** number : 10,000
- ***Vocabulary words*** number : 375,000
- Cover topics number : 20
- Cover emotions number : 8
- Average topics per document : 7
- Average emotions per document : 4


1. Introduction
2. Literature Reviews
3. Major Research Thomas
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichments Software Ecosystem
2. Semantic Metadata Enrichments: SMESE V3 (2/3)
ÉTS
1. Topics detection: BM-SSEA Algorithm
Sentences
Persons
Concepts
BM-SATD produced an average accuracy of 90% for one detected topic and 61% for ten topics (detected) topics compared to 80.29% and 41% (ten topics) for LDA-IG.
Average accuracy
Ave_acc
BM-SATD
LDA-IG
KeyGraph
LDA
HLTM
Number of detected topics
Average accuracy
2. Sentiment and emotion detection : BM-SATD Algorithm
Sentiments et émotions
BM-SSEA has an average accuracy of 93% per emotion while ETM-LDA, the best compared to the other two approaches, produced 69% per emotion.
Average accuracy
Ave_acc
BM-SSEA
ETM-LDA
AP
Number of discovered emotions
Average accuracy
43


1. Introduction
2. Literature Reviews
3. Major Research Thomas
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichments Software Ecosystem
Published articles related to this thesis
ÉTS
1. Software Ecosystem Models
2. Semantic Metadata Enrichments
3. Assisted Literature Reviews
A Semantic Metadata Enrichment Software Ecosystem (SMESE) Based on a Multi-Platform Metadata Model for Digital Libraries
A Semantic Metadata Enrichment Software Ecosystem based on Sentiment and Emotion Metadata Enrichments
Efficient Scientific Research Literature Ranking Model based on Text and Data Mining Technique
A Semantic Metadata Enrichment Software Ecosystem based on Metadata and Affinity Models
A Semantic Metadata Enrichment Software Ecosystem based on Topic Metadata Enrichments
TEXT AND DATA MINING & MACHINE LEARNING MODELS TO BUILD AN ASSISTED LITERATURE REVIEW WITH RELEVANT PAPERS
44

1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions

A Semantic Metadata Enrichments Software Ecosystem

3. Assisted Literature Reviews: Paper selection

ÉTS

	Authors	Algorithms	Positive	Negative
1	Bornmann et al., 2014, 2015			• Ranking des institutions au lieu des articles
2	Wan & Liu, 2014			• Authors Ranking instead of papers • Copie de h-index, Citations only • Undefined Feature
3	Hasson et al., 2014	Paper Time Ranking Algorithm (PTRA)	• Ranking des articles • Utilise 3 métadonnées (feature) • Combine citation-based et text-based	• Not assisted • Limité au ranking
4	Rúbio & Gulo, 2016	MLM ID3	• Ranking des articles • Utilise l'apprentissage machine (MLM) • Combine citation-based et text-based • Utilise 4 métadonnées (feature)	• Not assisted • Limité au ranking
5	Madani & Weber, 2016	eigenvector centrality	• Ranking des articles	• Il se limite au texte de l'abstract • Utilise un algo qu'il n'a pas conçu • Not assisted
6	Wang et al., 2014	MRFRank	• Ranking des articles • Extraction des caracté	• Il se limite au texte de l'abstract • Feature non défini, Not assisted
7	Gulo et al., 2015	Naive Bayes	• Ranking des articles • Utilise l'apprentissage machine (MLM)	• Not assisted • Limited ranking, Undefined Feature
8	- Z. He et al., 2015 - Fang et al., 2015 - CELEBI & DOKUN,2015 - Premjith, & Al., 2015 - (Mendoza & Al., 2014			• Résumé de texte pas applicable aux articles scientifiques • Pas de ranking
9	- Ronzano & Saggion, 2016			• Limited to summary of papers • No ranking

1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions

A Semantic Metadata Enrichments Software Ecosystem

3. Assisted Literature Reviews: Paper selection

ÉTS

	Authors	Algorithms	Positive	Negative
1	Bornmann et al., 2014, 2015			• Ranking des institutions au lieu des articles
2	Wan & Liu, 2014			• Authors Ranking instead of papers • Copie de h-index, Citations only
3	Hasson et			
4	Rúbio & G			
5	Madani &			
6	Wang et al			
7	Gulo et al., 2015	Naive Bayes	• Ranking des articles • Utilise l'apprentissage machine (MLM)	• Not assisted • Limited ranking, Undefined Feature
8	- Z. He et al., 2015 - Fang et al., 2015 - CELEBI & DOKUN,2015 - Premjith, & Al., 2015 - (Mendoza & Al., 2014			• Résumé de texte pas applicable aux articles scientifiques • Pas de ranking
9	- Ronzano & Saggion, 2016			• Limited to summary of papers • No ranking

39

Assisted Literature Review

Selection criteria:

Recent

Including Paper Ranking

Used techniques as Citation-based, Text-based et MLM

Used more metadata (feature) for the Ranking

PTRA uses 3 metadata (feature)

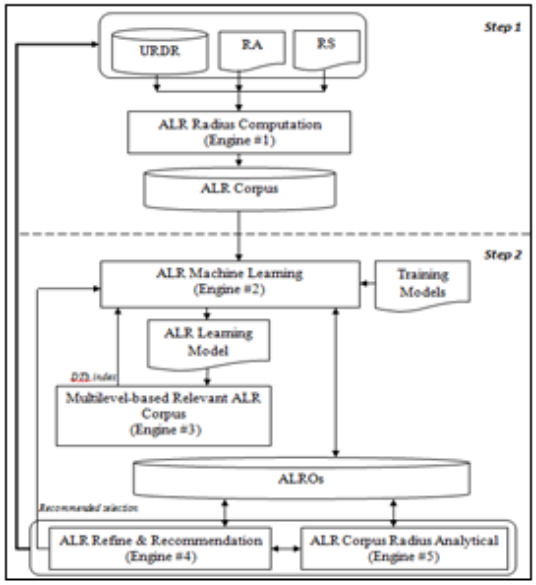
ID3 uses 4 metadata (feature)

A Semantic Metadata Enrichments Software Ecosystem

3. Assisted Literature Reviews: STELLAR V1 (1/3)

The contributions of STELLAR V1 are:

1. STELLAR proposed a new ***model and processes***.
2. New algorithms for identification and ranking of ***relevant papers based on multiple metadata***:
   (1) researcher metadata selection
   (2) age of papers
   (3) social-level metric and citation category
   (4) polarity to measure paper impact
   (5) others



A Semantic Metadata Enrichments Software Ecosystem

3. Assisted Literature Reviews: STELLAR V1 (3/3)

✓ Goal = Assist researchers in the evaluation of scientific papers relevancy, semantic similarity and ranking by topic or area of knowledge

The results of STELLAR:

Precision : Papers classified as relevant when they really are
Accurancy : Also included papers classified as non-relevant when they are not relevant

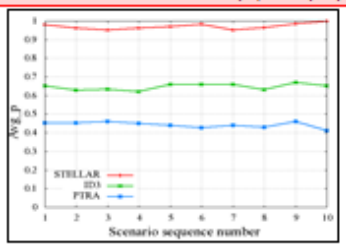
1. Baseline dataset #1 (58 papers of this thesis literature review – Cont#3)

Approaches	Year of publication	Citation number	Reference	Venue type	Venue age	Authors' impact	Citation category	Venue impact	Authors' institutes	Citing document	Researcher selection	Citing document of cited documents
PTRA (Hasson et al., 2014)	X	X		X								
ID3 (Rúbio & Gulo, 2016)	X	X	X	X								
STELLAR	X	X	X	X	X	X	X	X	X	X	X	X

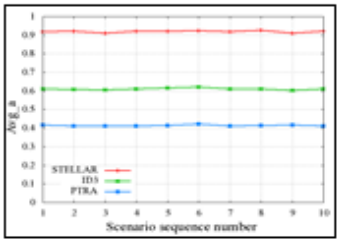
Approach	Avg_a (%)	Avg_p (%)
PTRA (Hasson et al., 2014)	39.19	27.16
ID3 (Rúbio & Gulo, 2016)	53.98	41.97
STELLAR	76.09	68.73

Average Relevancy and NOT (Avg_a) of **76%** while ID3 produced **54%**
Average Relevancy (Avg_p) of **69%** while ID3 produced **42%**

2. Harvested datasets #2 (2,000 papers)



Average Relevancy and NOT



Average Relevancy

A Semantic Metadata Enrichments Software Ecosystem

3. Assisted Literature Reviews: Scenario Sequence Number

ÉTS

The prototypes of STELLAR:

Scenario sequence number #1: (*discipline_1, language, title_1, topic_1, keywords_1, abstract_1*)
Scenario sequence number #2: (*discipline_2, language, title_2, topic_2, keywords_2, abstract_2*)
...

A. Baseline dataset

Scenario: **one simulator**

Run where the 58 papers constituted the dataset #1 – ***Stellar Paper #3***.

B. Dataset harvested from databases

- Intelligence: 1000 papers
- Information Systems: 1000 papers

Scenario: **two simulator runs** (one on each domain dataset #2)

For each Scenario, different RA and RS were used

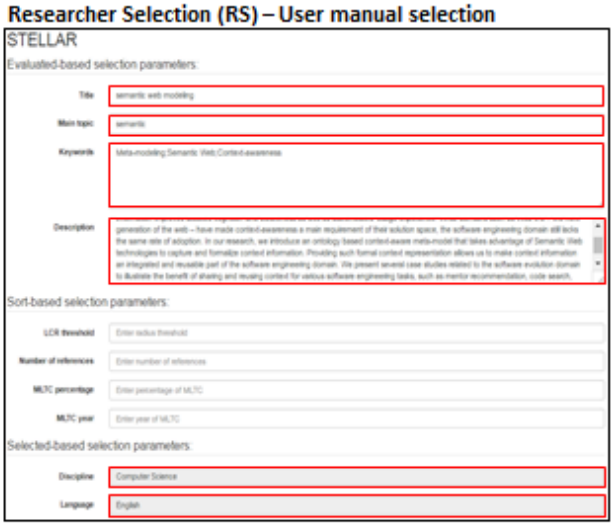
A Semantic Metadata Enrichments Software Ecosystem

3. Assisted Literature Reviews: Prototypes STELLAR (1/4)

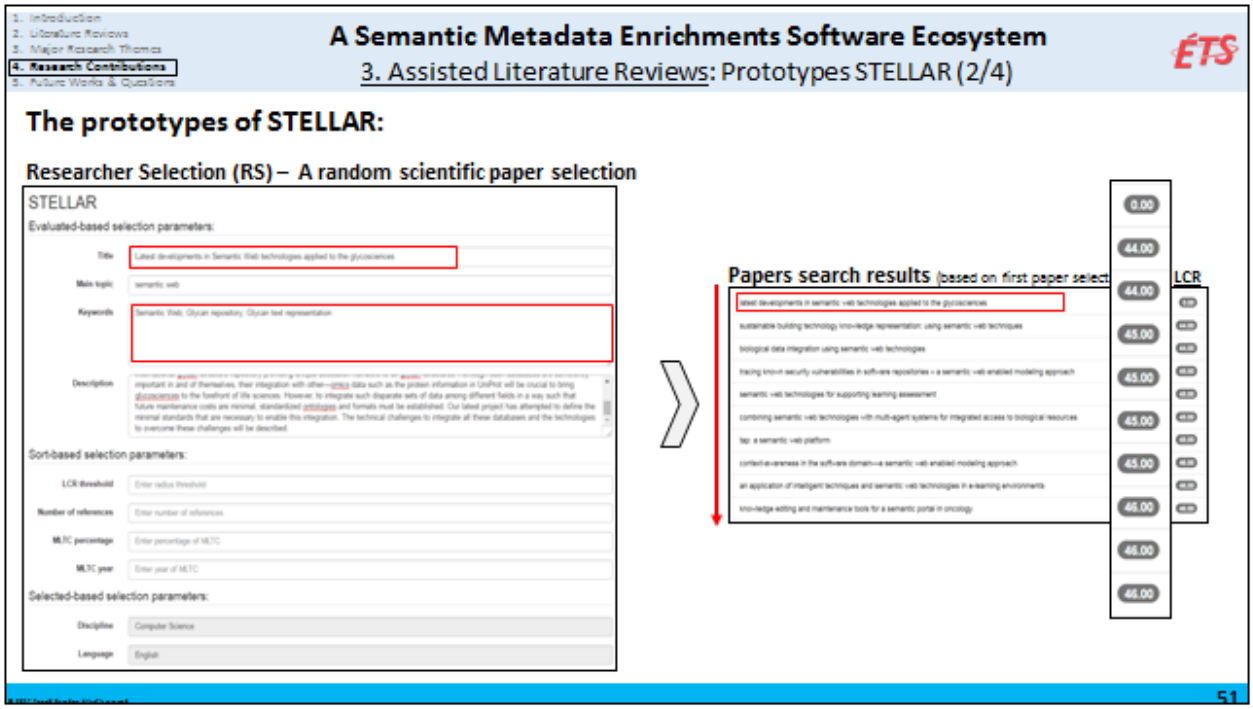
ÉTS

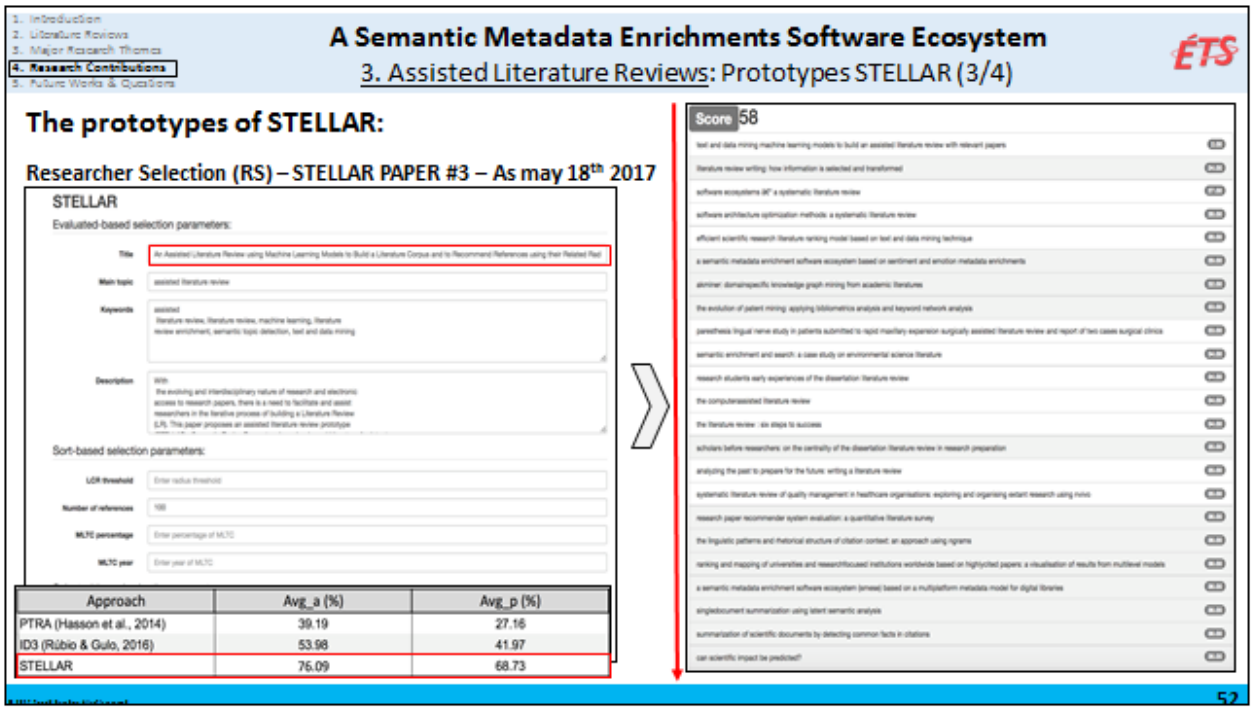
The prototypes of STELLAR – Uniqueness Model:

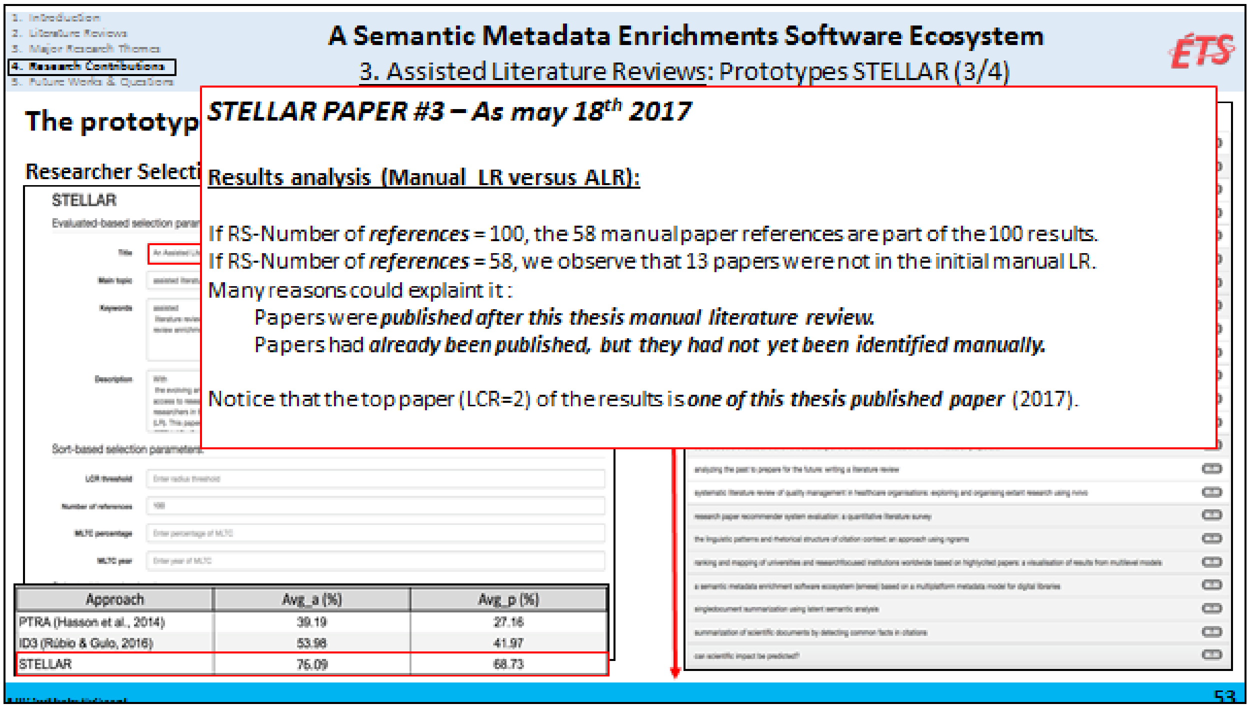
Researcher Selection (RS) – User manual selection



Papers search results (based on Researcher S


1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichments Software Ecosystem
3. Assisted Literature Reviews: Prototypes STELLAR (2/4)
ÉTS
The prototypes of STELLAR:
Researcher Selection (RS) – A random scientific paper selection
STELLAR
Evaluated-based selection parameters:
Sort-based selection parameters:
Selected-based selection parameters:
Papers search results (based on first paper select
LCR
51


1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichments Software Ecosystem
3. Assisted Literature Reviews: Prototypes STELLAR (3/4)
ÉTS
The prototypes of STELLAR:
Researcher Selection (RS) – STELLAR PAPER #3 – As may 18th 2017
STELLAR
Evaluated-based selection parameters:
Sort-based selection parameters:
Approach | Avg_a (%) | Avg_p (%)
PTRA (Hasson et al., 2014) | 39.19 | 27.16
ID3 (Rúbio & Gulo, 2016) | 53.98 | 41.97
STELLAR | 76.09 | 68.73
Score 58
52



1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions

A Semantic Metadata Enrichments Software Ecosystem

3. Assisted Literature Reviews: Prototypes STELLAR (3/4)

The prototyp

Researcher Selecti

STELLAR

Evaluated-based selection param

Title

Main topic

Keywords

Description

Sort-based selection parameters:

LCR threshold: Enter radius threshold

Number of references: 100

MLTC percentage: Enter percentage of MLTC

MLTC year: Enter year of MLTC

STELLAR PAPER #3 – As may 18th 2017

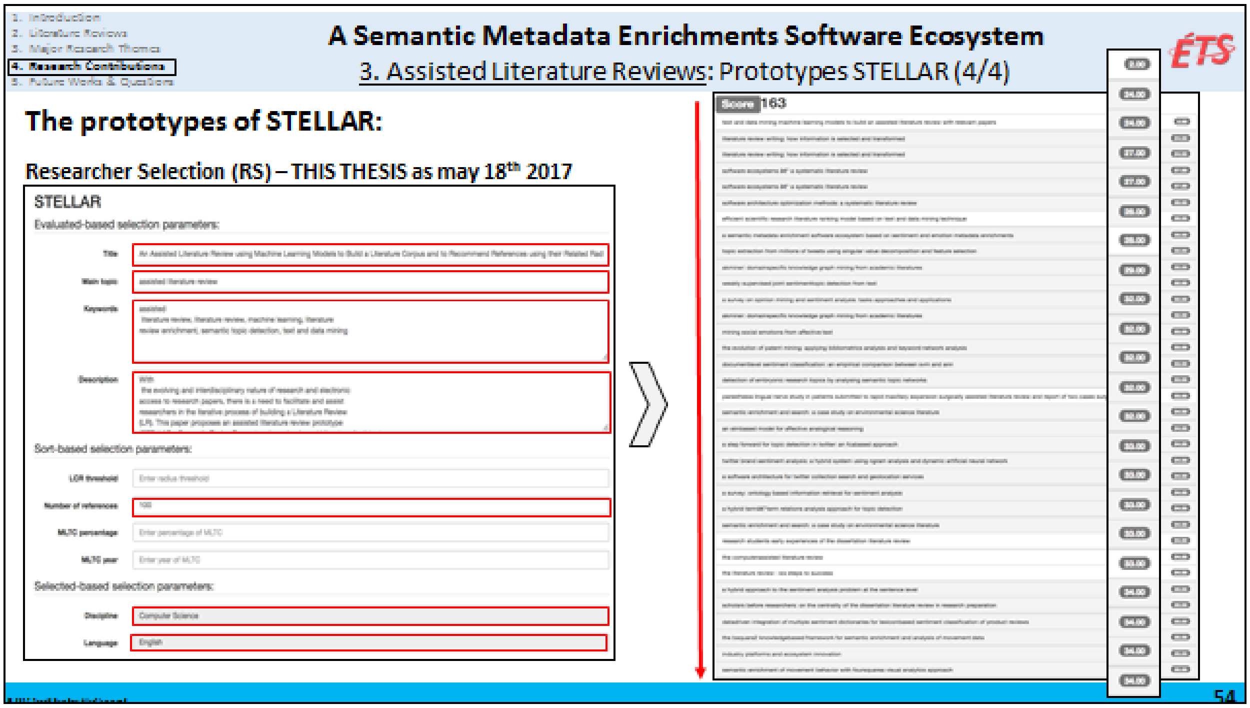
Results analysis (Manual LR versus ALR):

If RS-Number of ***references*** = 100, the 58 manual paper references are part of the 100 results.
If RS-Number of ***references*** = 58, we observe that 13 papers were not in the initial manual LR.
Many reasons could explain it :
Papers were ***published after this thesis manual literature review.***
Papers had ***already been published, but they had not yet been identified manually.***

Notice that the top paper (LCR=2) of the results is ***one of this thesis published paper*** (2017).

Approach	Avg_a (%)	Avg_p (%)
PTRA (Hasson et al., 2014)	39.19	27.16
ID3 (Rúbio & Gulo, 2016)	53.98	41.97
STELLAR	76.09	68.73

53



1. Introduction
2. Literature Reviews
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions

A Semantic Metadata Enrichments Software Ecosystem

3. Assisted Literature Reviews: Prototypes STELLAR (4/4)

The prototypes of STELLAR:

Researcher Selection (RS) – THIS THESIS as may 18th 2017

STELLAR

Evaluated-based selection parameters:

Title: An Assisted Literature Review using Machine Learning Models to Build a Literature Corpus and to Recommend References using their Related Rad

Main topic: assisted literature review

Keywords: assisted literature review, literature review, machine learning, literature review enrichment, semantic topic detection, text and data mining

Description: With the evolving and interdisciplinary nature of research and electronic access to research papers, there is a need to facilitate and assist researchers in the iterative process of building a Literature Review (LR). This paper proposes an assisted literature review prototype

Sort-based selection parameters:

LCR threshold: Enter radius threshold

Number of references: 100

MLTC percentage: Enter percentage of MLTC

MLTC year: Enter year of MLTC

Selected-based selection parameters:

Discipline: Computer Science

Language: English

Score 163

54

A Semantic Metadata Enrichments Software Ecosystem

ÉTS

THIS THESIS – As may 18th 2017

Results analysis (Manual LR versus ALR):

If RS-Number of **references = 250**, all of the references appear in the list of results.

If RS-Number of **references = 163** such as the number of references of the thesis, there are just 12 references which should not have been cited; that provides an accuracy ***of 93%.***

In these 12 forgotten references,

we added 4 new of the 7 papers published from this thesis:

- 2 ***have been published in 2017***; so after the literature review
- 4 ***have been published in 2017***; from this thesis (3 were already in the papers reference list)
- 2 ***have been published in 2014 and 2015***; they had not been identified during papers manual search
- 4 ***have been published between 1961 and 2009; despite their relevance, there are out of date.***

In the top 6 papers of the simulation results, **3 are the thesis published papers**;

55

A Semantic Metadata Enrichments Software Ecosystem

its Prototypes for Digital Libraries, Metadata Enrichments and Assisted Literature Reviews

ÉTS

1. Introduction

1. Context of the thesis (Motivations and Goals)
2. Overview of the thesis

2. Literature Reviews

1. Software Ecosystem Model
2. Semantic Metadata Enrichments
3. Assisted Literature Reviews

3. Major Research Themes

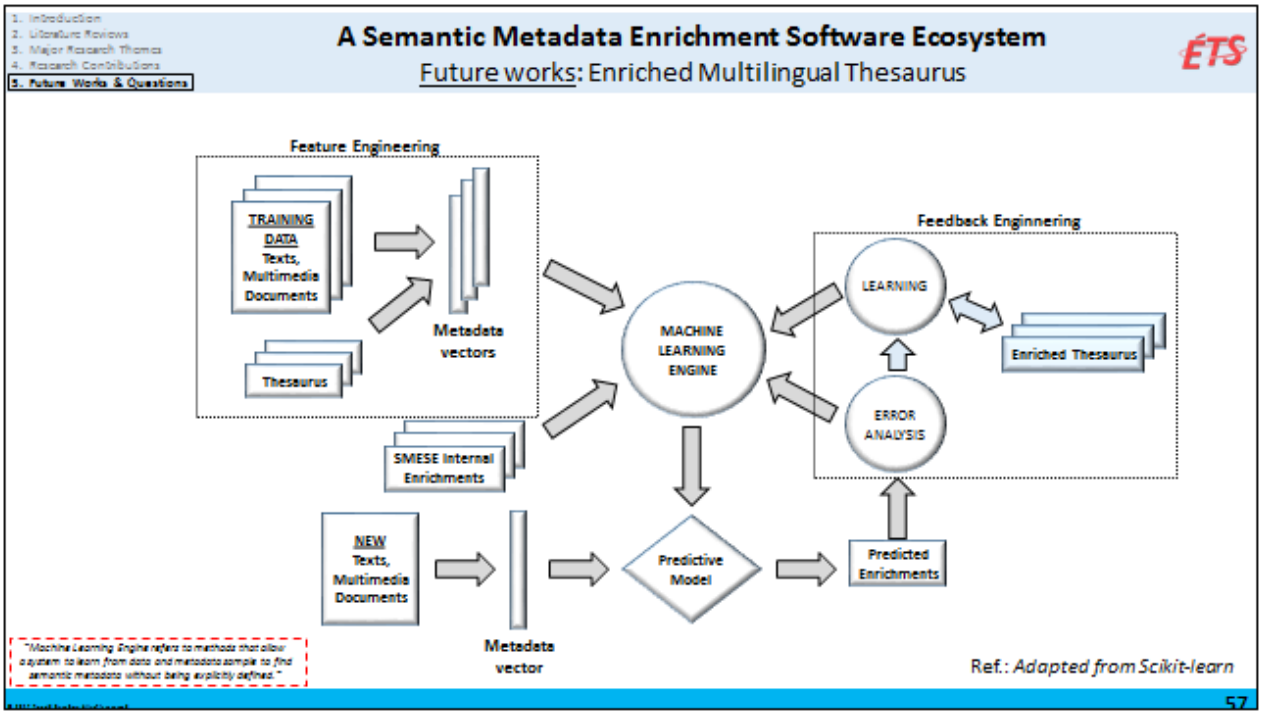
1. Software Ecosystem Model (SMESE V1)
2. Semantic Metadata Enrichments (SMESE V3)
3. Assisted Literature Reviews (STELLAR V1)

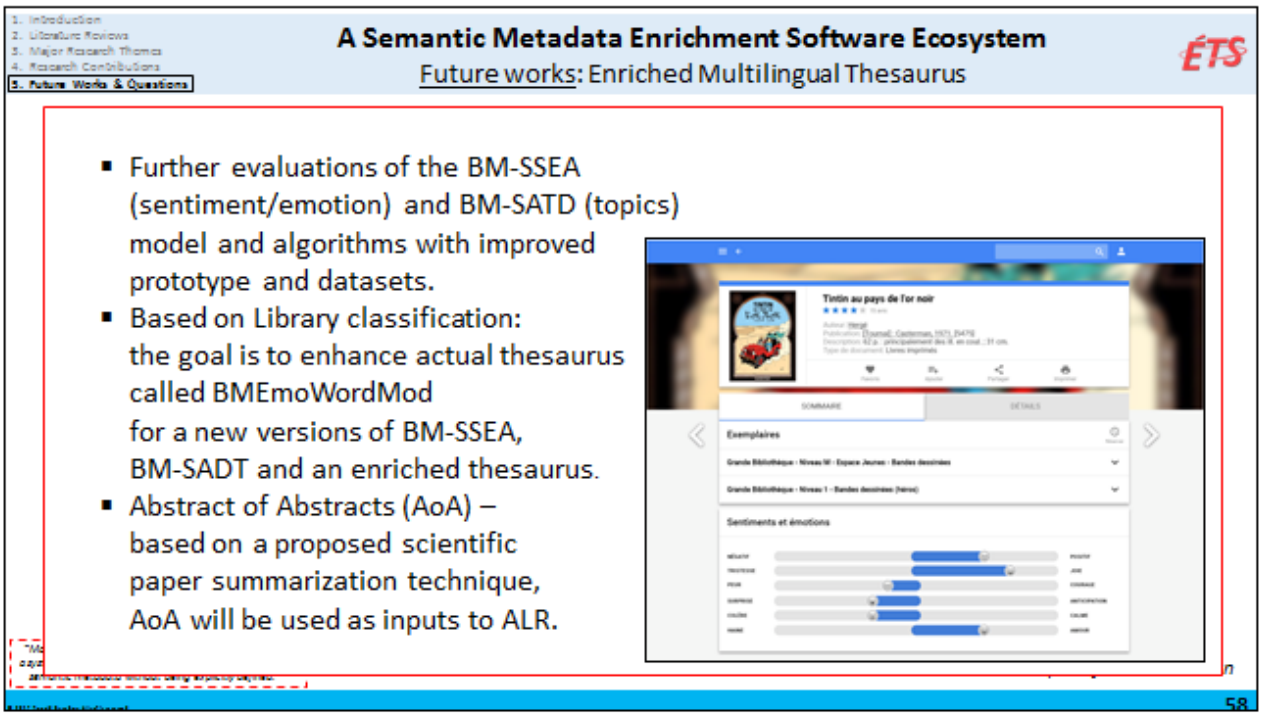
4. Research Contributions

1. Published articles related to this thesis
2. Software Ecosystem Models (SMESE V1)
3. Semantic Metadata Enrichments (SMESE V3)
4. Assisted Literature Reviews (STELLAR V1)

5. Future Works & Questions

56


1. Introduction
2. Literature Review
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichment Software Ecosystem
Future works: Enriched Multilingual Thesaurus
ÉTS
Feature Engineering
TRAINING DATA
Texts, Multimedia Documents
Thesaurus
Metadata vectors
MACHINE LEARNING ENGINE
Feedback Enginnering
LEARNING
Enriched Thesaurus
ERROR ANALYSIS
SMESE Internal Enrichments
NEW
Texts, Multimedia Documents
Metadata vector
Predictive Model
Predicted Enrichments
"Machine Learning Engine refers to methods that allow a system to learn from data and metadata sample to find semantic metadata without being explicitly defined."
Ref.: Adapted from Scikit-learn
57


1. Introduction
2. Literature Review
3. Major Research Themes
4. Research Contributions
5. Future Works & Questions
A Semantic Metadata Enrichment Software Ecosystem
Future works: Enriched Multilingual Thesaurus
ÉTS
Further evaluations of the BM-SSEA (sentiment/emotion) and BM-SATD (topics) model and algorithms with improved prototype and datasets.
Based on Library classification: the goal is to enhance actual thesaurus called BMEmoWordMod for a new versions of BM-SSEA, BM-SADT and an enriched thesaurus.
Abstract of Abstracts (AoA) – based on a proposed scientific paper summarization technique, AoA will be used as inputs to ALR.
Tintin au pays de l'or noir
Exemplaires
Sentiments et émotions
58

A Semantic Metadata Enrichments Software Ecosystem

its Prototypes for Digital Libraries, Metadata Enrichments and Assisted Literature Reviews

ÉTS

Thank you for your attention

Questions ?

Supplementary

A Semantic Metadata Enrichments Software Ecosystem

Published articles related to this thesis: 7 papers

ÉTS

Number	Paper Title	Journal	Impact Factor
Paper #1	A Semantic Metadata Enrichment Software Ecosystem (SMESE) based on a Multi-platform Metadata Model for Digital Libraries	Journal of Software Engineering and Applications (JSEA)	2-GJIF: 1.25 RGJI: 0.5 14th in the top 20 publications matching Software Engineering based on Google Scholar Metrics (June 2016)
Paper #2	A Semantic Metadata Enrichment Software Ecosystem based on Metadata and Affinity Models	International Journal of Information Technology and Computer Science (IJITCS)	GIF 2015: 0.716 ICV 2014: 8.31
Paper #3	A Semantic Metadata Enrichment Software Ecosystem based on Sentiment and Emotion Metadata Enrichments	International Journal of Scientific Research in Science Engineering and Technology (IJSRSET)	SJIF 2015: 3.632 GIF 2015: 0.453
Paper #4	A Semantic Metadata Enrichment Software Ecosystem based on Topic Metadata Enrichments	International Journal of Data Mining & Knowledge Management Process (IJDKP)	
Paper #5	A Semantic Metadata Enrichment Software Ecosystem based on Machine Learning to Analyse Topic, Sentiment and Emotions	INTERNATIONAL JOURNAL OF RECENT SCIENTIFIC RESEARCH (IJRSR)	SJIF 2016: 6.86 ICV: 5.72
Paper #6	Efficient Scientific Research Literature Ranking Model based on Text and Data Mining Technique	International Journal of Engineering Research And Management (IJERM)	IF 2014-2015: 2.37
Paper #7	Text and Data Mining & Machine Learning Models to Build an Assisted Literature Review with Relevant Papers	International Journal of Scientific Research in Information Systems and Engineering (IJSRISE)	GIF 2015: 0.565

- 2-year Google-based Journal Impact Factor (2-GJIF)
- Scientific Journal Impact Factor (SJIF)
- Global Impact Factor (GIF)
- Index Copernicus Value (ICV)
- ResearchGate Journal Impact (RGJI)

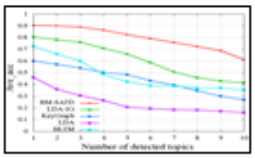
Supplementary

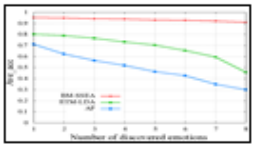
ÉTS

(1) Topics, (2) Emotions and sentiments

The average accuracy, Ave_acc, of multiple runs was given by:

$$Ave\_acc = \frac{\sum_{x=1}^{I}\left(\frac{\sum_{d \in TD} A_d^t}{|TD|}\right)}{I}$$





Where

- TD denotes the number of tests documents
- I denotes the number of test iterations
- A_d^t denotes the accuracy of topics detection

A_d^t, was computed as follows:

$$A_d^t = \frac{2x|T_{annotated} \cap T_{detected}|}{|T_{annotated}| + |T_{detected}|}$$

Where

- $T_{annotated}$ denotes the set of annotated (manual) topics for a given document d
- $T_{detected}$ denotes the set of detected (SMESE) topics by BM-SATD for a given document d
- $T_{annotated}$ denotes the set of annotated (manual) emotion/sentiment for a given document d
- $T_{detected}$ denotes the set of detected (SMESE) emotion/sentiment by BM-SSEA for a given document d

1. Introduction
2. Literature Reviews
3. Major Thesis
4. Contributions
5. Future Works & Questions

Supplementary

A Semantic Metadata Enrichments Software Ecosystem

Future works: STELLAR V2

ÉTS

Future Works:

- Recommended User Interest-based New Content and Event
  Its goal is to enrich different types of content metadata (books) with the related interests metadata.