Dealing with Missing Data and Data Fusion in Smart Environment Context

by

Andre Luis COSTA CARVALHO

THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
IN PARTIAL FULFILLMENT OF A MASTER'S DEGREE
WITH THESIS IN INFORMATION TECHNOLOGY ENGINEERING
M.A.Sc.

MONTREAL, FEBRUARY 12, 2021

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

Mr. Mohamed Cheriet, Thesis Supervisor
Department of Systems Engineering, École de technologie supérieure

Mr. Kim-Khoa Nguyen, President of the board of examiners
Department of Electrical Engineering, École de technologie supérieure

Mr. Mazdak Nik-Bakht, External examiner
Department of Building, Civil, and Environmental Engineering, Concordia University

THIS THESIS WAS PRESENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND THE PUBLIC

ON FEBRUARY 05, 2021

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGEMENTS

Firstly, I would like to express my gratitude to prof. Mohamed Cheriet, my supervisor and Ph.D. Eng. Darine Ameyed research associated at Synchromedia Lab. Their guidance and constant support for the papers and for this thesis were essential for my master's successful conclusion.

I also want to thank Prof. Kim Khoa Nguyen for his kind advises.

In addition to, I would like to thank the board of examiners for reviewing and agreeing to judge my thesis and my friends at Synchromedia for their great support, and Rafael Amado for the philosophical talks and insights as well.

Finally, I thank my wife Juliana and my daughter Isabela for their motivation. There are not enough words to express how grateful I am. I want to extend my sincere thanks to my family: my mother, my father, and siblings.

Traitement des données manquantes et fusion de données dans un contexte d'environnement intelligent

Andre Luis COSTA CARVALHO

RÉSUMÉ

L'afflux croissant de personnes vivant dans les mégapoles exige une approche intelligente pour créer une infrastructure durable dans les zones urbaines et fournir des services plus efficaces. Les bâtiments qui ouvrent automatiquement les portes d'entrée, les lumières s'allument, les systèmes de chauffage et de refroidissement s'adaptant d'eux-mêmes, les caméras surveillant le trafic, entre autres situations, sont des exemples d'innombrables réseaux de capteurs interagissant avec les espaces et les personnes. Ces capteurs et systèmes envoient un volume énorme de données dans la plate-forme logicielle des centaines de fois par minute, exigeant une vitesse élevée dans le traitement et le stockage de ces informations via le réseau, l'Internet des objets. Ce comportement numérique fonde le concept de ville intelligente. Si la variété, le volume et la vitesse des données sont la définition du Big Data, les deux sont ancrés dans le développement des technologies de l'information et de la communication.

Cependant, les données disponibles sont distribuées et agrégées collectivement et leur fusion pourrait révéler des tendances qui ne seraient pas possibles si les données étaient analysées séparément. L'hypothèse principale de l'examen de la fusion de données est que la vue d'ensemble des informations fusionnées permet d'optimiser le flux d'électricité à travers le réseau électrique, de soutenir le déplacement des réseaux de transport, de veiller à la santé et à la sécurité des personnes, et bien plus encore. Pourtant, selon la revue de la littérature, il existe deux problèmes majeurs liés aux données de fusion. Premièrement, dans un scénario réel, tout système (par exemple, une application et une plate-forme) est soumis à la production de données qui pourraient être imprécises, insuffisantes, dupliquées, incorrectes, incohérentes, ambiguës, en plus des valeurs manquantes. Deuxièmement, dans la plupart des cas, les solutions de fusion se concentrent sur une stratégie minière prédéfinie et des tâches supervisées.

Pour s'affranchir du premier problème, il est à noter que les valeurs manquantes peuvent affecter considérablement le résultat des analyses et de la prise de décision dans n'importe quel domaine et que les deux principales approches pour traiter ce problème sont des méthodes statistiques et basées sur des modèles. Alors que la première apporte un biais aux analyses, la seconde est généralement conçue pour des cas spécifiques. Pour faire face aux limites des deux méthodes, nous présentons un cadre d'ensemble empilé basé sur l'intégration de l'algorithme de forêt aléatoire adaptative, de l'indice de Jaccard et de la probabilité bayésienne. Compte tenu du défi que représentent les données hétérogènes et distribuées provenant de sources multiples, nous avons construit un modèle d'utilisation qui prend en charge différents types de données: continues, discrètes, catégorielles et binaires.

Dans le but de surmonter la deuxième limitation de la fusion de données, nous introduisons un modèle d'apprentissage d'ensemble de fusion pour des ensembles de données multiples et hétérogènes empilant la machine Boltzmann restreinte, qui rassemblent les caractéristiques latentes des ensembles de données non étiquetés et l'algorithme de tri-factorisation matricielle pour fusionner les caractéristiques dans une structure de matrice de blocs. En combinant ces techniques, nous avons pu concevoir un outil efficace de découverte de connaissances. L'évaluation des deux solutions proposées tient compte de l'imputation des valeurs manquantes, et les données de fusion ont montré que notre modèle d'apprentissage d'ensemble produit des résultats prometteur et compétitifs, surmontant les limites précédemment décriteo dans la revue de la littérature.

Mots-clés: Big data, fusion de données, apprentissage en profondeur, apprentissage d'ensemble, ville intelligente, imputation de données manquantes, données distribuées, multi domaines

Dealing with Missing Data and Data Fusion in Smart Environment Context

Andre Luis COSTA CARVALHO

ABSTRACT

The rising inflow of people living in megacities has demanded a smart approach to create a sustainable infrastructure to urban areas and provide more efficient services. Buildings automatically opening front doors, lights flicking on, heating and cooling systems adjusting by themselves, cameras tracking the traffic among other situations, are examples of countless arrays of sensors interacting with spaces and people. These sensors and systems dispatch huge volume of data into software platform hundreds of times per minute, demanding high velocity in processing and storing this information through the network, Internet of Things. This digital behaviour gives a foundation to the concept of smart city. While the data variety, volume and velocity are the Big Data definition, both are anchored on the development of Information and Communication Technology.

However, the available data are distributed and collectively aggregated and its fusion might reveal patterns that would not be possible if the data were analyzed separately. The main assumption in data fusion review is that the big picture from fused information allows the optimization of electricity flow through the power grid, supporting transportation networks moving, watching over people's health and safety, and much more. Yet, according to the literature review, there are two major problems related to fusion data. The first issue is, any system (e.g., application and platform) is susceptible to produce data that might be inaccurate, insufficient, duplicated, incorrect, inconsistent, ambiguous, besides the missing values. The second, in most cases, fusion solutions focus on predefined mining strategy and supervised tasks.

To overcome the first issue, it is noted that missing values can significantly affect the result of analyses and decision making in any field and that the two major approaches to deal with this issue are statistical and model-based methods. Whereas the former brings bias to the analyses, the latter is usually designed for specific cases. To cope with the limitations of both methods, we present a stacked ensemble framework integrating the adaptive random forest algorithm, the Jaccard index, and Bayesian probability. Considering the challenge that the heterogeneous and distributed data from multiple sources represents, we have built a model that supports different data types: continuous, discrete, categorical, and binary.

Aiming to overcome the second limitation of data fusion, we introduce a fusion ensemble learning model for multiple and heterogeneous datasets stacking the Restricted Boltzmann Machine, which gather latent features of the unlabelled datasets and the matrix tri-factorization algorithm to fuse the features in a block-matrix structure. Combining such techniques, we were able to design an efficient knowledge discovery tool. The evaluation of both proposed solutions, missing values imputation and fusion data has shown that our ensemble learning model produces encouraging and competitive results, overcoming the limitations previously found in the literature review.

Keywords: Big data, Data fusion, Deep Learning, Ensemble learning, Smart city, Missing data imputation, Distributed data, Multidomains

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

DESA	Department of Economic and Social Affairs
UN	United Nations
ICT	Information and Communication Technology
IoT	Internet of Things
JDL	Joint Directors of Laboratories
MAR	Missing At Random
MCAR	Missing Compleat At Random
MNAR	Missing Not At Random
MFV	Most Frequent Value
PCA	Principal Componet Analysis
RBM	Restricted Boltzmann Machine
KDD	Knowledge Data Discovery
ML	Machine Learning
DM	Data Mining
RF	Random Forest
NLP	Natural Language Processing
RDBMS	Relational Data Base Management System
RDD	Resilient Distributes Dataset
ANN	Artificial Neural Network

GLRM	Generalization Low Rank Model
DNN	Distributed Neural Network
IID	Identically Independently Distributed
DF	Data Fusion
DFMF	Data Fusion Matrix Factorization

INTRODUCTION

This introductory chapter's purpose is to present the motivation and importance of research on data quality, focusing on missing values and their impact on data fusion problems. We provide a background of these topics and how they are related to smart cities and their future success.

0.1 Context and motivation

By 2050, about 70% of the world's population will be living in urban areas. However, according to, Department of Economic and Social Affairs (2019) United Nations (DESA-UN), for the first time in human history, the urban and rural population was equal in 2008. As UN demographers reported in 2008, the other two landmarks threshold were reached besides this immigration to the cities' phenomena. The number of users accessing the internet via wireless has exceeded the piped user connections, and the total amount of people connected to become a minority in relation to the connected devices. The internet of people has been beaten by the Internet of things and both events have been outlining important challenges in the present and into the upcoming future.

Cities worldwide have been employing Information and Communication Technologies (ICT) in response to the rising inflow of people living in megacities, which has demanded a smart approach to create an infrastructure to making urban areas more sustainable and provide services more efficiently. In the last decade, following this ongoing urbanization process, advances such as the Internet of Things (IoT) and Big Data have become the pillars of smartness concepts, Marjani, Nasaruddin, Gani, Karim, Hashem, Siddiqa & Yaqoob (2017). Smart spaces, including smart cities, are digital and physical environments where humans and technology interact coordinately with each other. It aims to offer a collaborative-intelligent approach that holds together services, processes, and data necessary to an individual's or community's activities, combining smart and synchronized solutions across the targeted ecosystem, Cearly (2019); Qin & Gu (2011).

Nevertheless, achieving an integrated smart ecosystem remains challenging because of existing issues as data quality, variety, interoperability and so on. Systems platforms have been producing great volume of data in different domains including security, mobility, education, and health. A cross-domain analysis will be able to provide a more efficient service. The data fusion will allow complete, accurate and rapid evaluation of the cities' services. Plus, the ability to fuse data enables context awareness which has a huge potential to enhance the efficiency of the urban space management, make faster evaluation of the efficiencies of city services, and provides vital knowledge back to the citizens.

Still, a critical aspect of the data that shapes a smart city is the distribution from different domains, in Shahat, Elragal & Bergvall-Kareborn (2017). Typically, data in smart cities are collected from different sources, under distinct formats and usually composed of unstructured and semi-structured data (e.g., images, streaming, and text files), as briefly summarized in Fig. 0.1. Therefore, dealing with distributed, multiple sources and formats, and ensuring data quality is essential to improving the results of the fusion process which can enhance the learning and knowledge extraction process, revealing the service quality and, ultimately, increasing comfort and well-being of the citizens, Lau, Marakkalage, Zhou, Hassan, Yuen, Zhang & Tan (2019).

In this context data fusion still represents multiple challenges such as heterogeneity, interoperability, and other issues previously mentioned. When applied in smart cities' context, data fusion means a multi-level process focused on improving the awareness of public spaces taking into account multiple sources of data, Hall & Llinas (1997); Castanedo (2013); Zitnik & Zupan (2015).

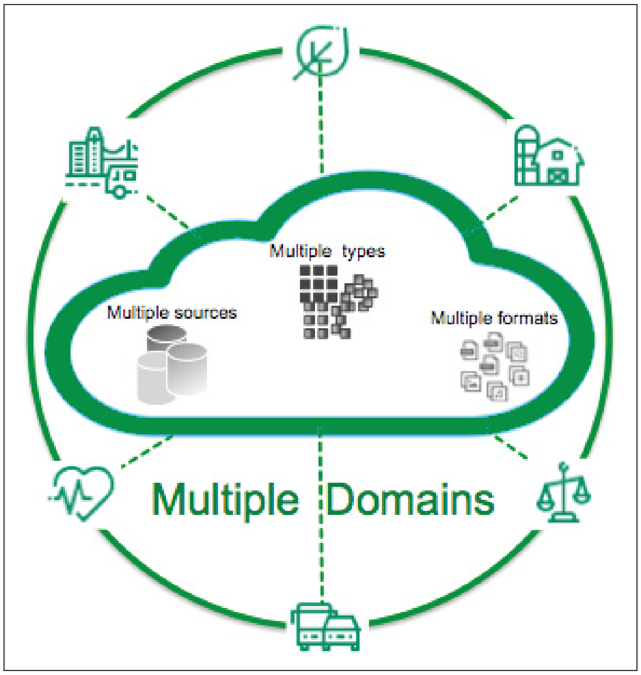


Figure 0.1 Distributed data

0.2 Problem statement

In the literature, the are three prominent data fusion definitions, as follows: The Joint Directors of Laboratories (JDL) defines data fusion as the process in which data are associated, correlated and combined, Hall & Llinas (1997). In, Durrant-Whyte & Hugh (1990), the fusion technique depends on the relation of the input data. This data should be complementary, redundant, or cooperative. A classification for the fusion process in terms of input data and the output results are explained in, Dasarathy (1997). By all means, the fusion process is a data quality problem dependent.

Thereby, there are two main problems of data fusion: The first is regarding the quality of the fused data. Data quality may directly impact the fusion process because although the information can be verified, they cannot ensure its quality. The shortage of validation results in insufficient, duplicated, ambiguous, missing, and many other data problems, which can be even worse in a distributed scenario, Lau *et al.* (2019). Missing data is a widespread challenge that affects the quality of any study and produce biased estimates, leading to invalid conclusions. In fact, any system (e.g., application, and platform) is subject to produce incomplete data, Hatem (2017). Considering analytics and learning processes, such inconsistency can significantly impact the conclusions drawn from the data and may bias the decision process. Health, all sorts of businesses, surveys of any nature, and automated sensors are just a few of countless examples, Mohan & Pearl (2018), Azimi, Pahikkala, Rahmani, Niela-Vilén, Axelin & Liljeberg (2019) of areas that might be impacted by such a problem.

In the literature, one of the possible reasons for the data loss is missing data, and adequate treatment is usually in accordance with three prominent techniques:

- *missing at random (MAR)*: the probability that an attribute will have missing values depends on another attribute or observation.
- *missing completely at random (MCAR)*: all attributes will have the same probability of missing values independent of the observation.
- *missing not at random (MNAR)*: the missing value in an attribute is strictly dependent on the values in other attributes or observations, Schafer & Graham (2002).

In Fig. 0.2 is shown a sample of the three cases. Over the past years, the missing values problem has been tackled mostly from either a statistical or ad-hoc model-based perspective.

The second problem is to determine the best fusion procedure according to the heterogeneous datasets that are the inputs of the process. Fig. 0.3 illustrates such dependency of the data fusion process,(Dasarathy (1997)) according to the following input and output data types: Data in/Data out, Data in/Feature out, Feature in/Feature out, Feature in/Decision out, Decision in/Decision out. The levels, data, feature, or decision employed in the fusion process depends on which data source an individual has in hands. Eventually, more than one level must be associated to achieve the primary goal, Pavlidis *et al.* (2002). In this scenario, despite numerous and distinct

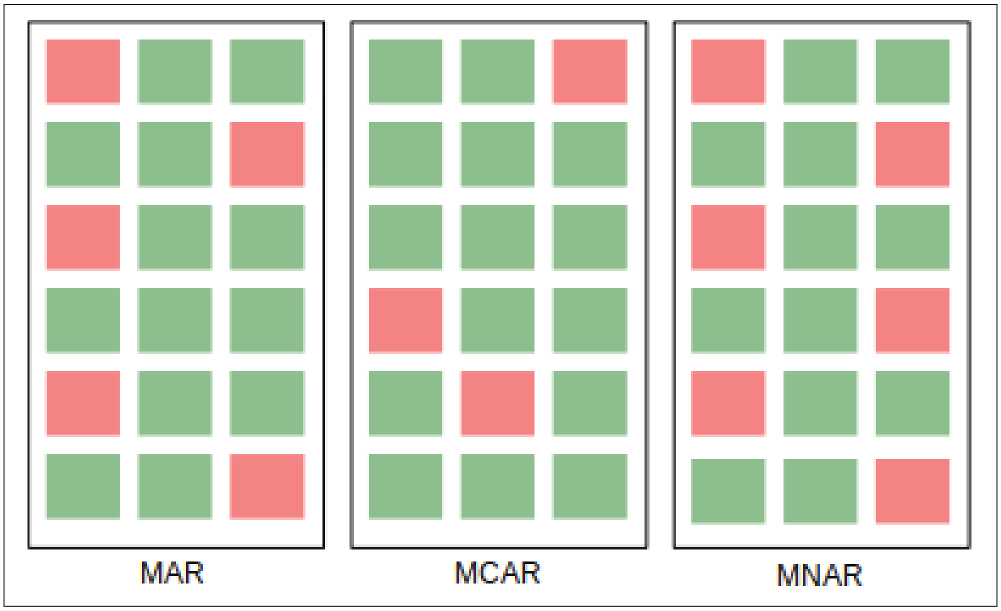


Figure 0.2 Missing data cases

techniques applied in the process, data heterogeneity, such as text, streaming or geospatial data, is gathered from different sources and domains. The data should be aligned before being fused. Its dimensionalities must be commonly framed among the datasets, Meng, Jing, Yan & Pedrycz (2020), and the outcomes of the fusion process are highly dependent on such choice, Tan, Zhang, Mao & Qian (2015).

0.3 Research questions

The two exposed problems, the reason for this thesis, are in summary:

1. missing data imputation.
2. data fusion in heterogeneous datasets.

To properly address these two issues, we consider the logical dependency between them, when reconstructing the datasets that suffer from missing values, which will enhance the quality of the fusion process; in this work, we elaborated two research question for the first problem and two

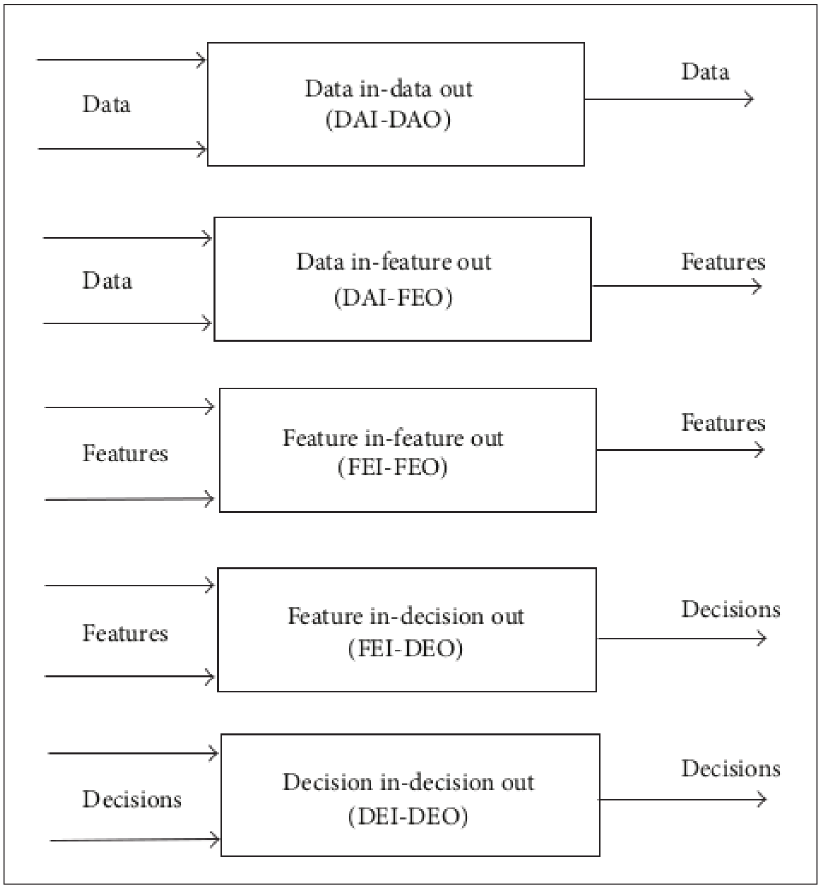


Figure 0.3 Dasarathy's classification
Taken from Castanedo (2013)

research questions for the second, respectively:

- **RQ1.** How can missing values be inferred to enhance the power of analysis and quality results of a dataset, which might have had its capacity degraded due to the missing values problem?

  A common problem in open and multiple source datasets is that data are generated by different systems, databases, and platforms. Although the sources can be verified, they cannot ensure the quality of the data. Also, problems like insufficient, duplicated, incorrect, uncertain, ambiguous, and mainly missing values are frequently present.

- **RQ2.** Is it possible to use model-based methods to infer missing values considering the portion of the dataset that does not suffer from this problem?

  Machine learning is the most up-to-date approach to tackle the problem of missing values whereas statistical techniques are the old fashion manner. Statistical methods such as mean, median, mode, and so on, can impute missing values. However, such methods might lead to undesirable outcomes, such as a) distortion of the original variance, b) distortion of covariance and correlation with other columns within the dataset, and c) overrepresentation of the MFV when a large number of observations are missing, Schafer & Graham (2002).

- **RQ3.** How can a fusion process be structured to deal with heterogeneous datasets with different dimensions?

  In order to use the information in the datasets as much as possible, the first step is to consider an algorithm able to resize it in the latent features level. It should be considered that heterogeneous datasets have different dimensions. For example, we have a dataset *A* in a *m X n* format, a dataset *B* in a *j X k* format. If dimension $n \neq k$, somehow, this difference must be adjusted because the matrix factorization is a linear algorithm.

- **RQ4.** How fusion alignment can be performed considering the heterogeneity in the dataset with different structures?

  Semantic alignment is necessary when the sources of data do not refer to a common object or phenomena. Otherwise, if the inputs come from the same sensors' type, the semantic alignment is not necessary to observe the same object or phenomena. It is typically demanded in the fusing processes of many different datasets. In several cases, different sensors use either different set of names or different symbols for the same phenomena. In this thesis, our framework's performance is analyzed by fusing geospatial datasets that relate various services provided in the city of Montreal open data platform.

0.4 Thesis objectives

The work's main objective is to propose a framework that fuses heterogeneous data, aiming to perform knowledge data discovery (KDD). However, considering that dealing with the critical missing data problem found in the real-world datasets is a key factor to an effective and efficient fusion. More specifically, our sub-objectives are as follows:

- **SO1.** To build a framework able to reconstruct datasets that suffer from missing values, applying an ensemble machine learning approach instead of the techniques such as mean, median, or mode, already described in the data quality fields.
- **SO2.** To build a framework able to fuse multiple heterogeneous datasets. The proposed data fusion model, intends to using the datasets available on the ICT infrastructure of Montreal as use-case. These datasets cover different aspects (social, environmental, and economic) of the town and they have been collected from different layers and services with a purpose of understanding how they correlate based on an ensemble unsupervised machine learning model.

0.5 Main contributions

Our main contributions have been divided in two parts, summarized as follows:

Regarding missing values

Taking into consideration the limitations found in statistical techniques and narrowed model-based tools to attend the missing values problem, we proposed an ensemble learning missing data imputation model to actuate in the preprocessing phase of the fusion. It is a key step for coherent results. The proposed framework is able to perform in data types such as numeric, categorical, and boolean in labelled and unlabeled datasets.

- it increases the quality of datasets affected by missing values. This is a basic problem faced daily while preprocessing the data in ML pipeline.
- it decreases the level of lost information in the cleaning phase, producing better results for the further steps of data manipulation in both ML and DM processes.
- this framework saves time and effort in data cleaning and data preparation, often a time-consuming step in the ML and DM process.
- lastly, it adapts an RF algorithm that can store different subsets of dataset which can be used to impute the missing values.

Regarding data fusion

In literature review the PCA is the most common technique applied in the reduction case. However, in our proposed ensemble framework model, we investigated Restricted Boltzmann Machine (RBM) due to its capability to either produce dimensionality reduction as PCA does or increase the dimension's size, which is not possible with PCA.

- we match Restricted Boltzmann Machine (RBM) and Natural Language Processing (NLP) to capture most relevant features from multiple, heterogeneous, and unlabelled datasets that are going to be fused. Performing this feature learning in non-text and text data to fuse only significant features in a block-based matrix tri-factorization (MF) process results in a comprehensive framework able to work on multi-modal datasets.
- a reproducibility module is created and can recognize the significant features of the datasets and disregard features which do not contribute to the fusion outcomes.

The contributions highlighted above resulted in two conference publications.

0.6 Thesis outline

The thesis is divided in four main chapters. The introductory chapter presents the context and motivation of this study, defines the problem statement, the research questions, the objectives, and express our main contributions. The remaining chapters are as follows:

- the first chapter presents the survey and literature review, the key datasets concepts, such as dataset heterogeneity, distributed, and validation. Also, we define the background of the missing values problem and its most common cases. In addition we presents how this problem is tackled according to the statistical and model-based perspective. Data fusion classification is also presented and we discuss why data quality is a critical factor in the fusion process. Finally, we present the techniques for missing values imputation and data fusion.
- the second chapter presents the overall methodology, where data quality is discussed, focusing on missing values problem. A missing values imputation framework is proposed, and its structure is explained. Also, we propose a data fusion framework, and its main phases are described.
- the third chapter contains the experiments and validation results subdivided in part I and II. In first part gives the evaluation and results of the proposed missing values imputation framework. In the second part, we report the outcomes of the Data fusion framework.
- the fourth chapter discusses the strengths and weaknesses of both proposed frameworks and highlights the future research direction.

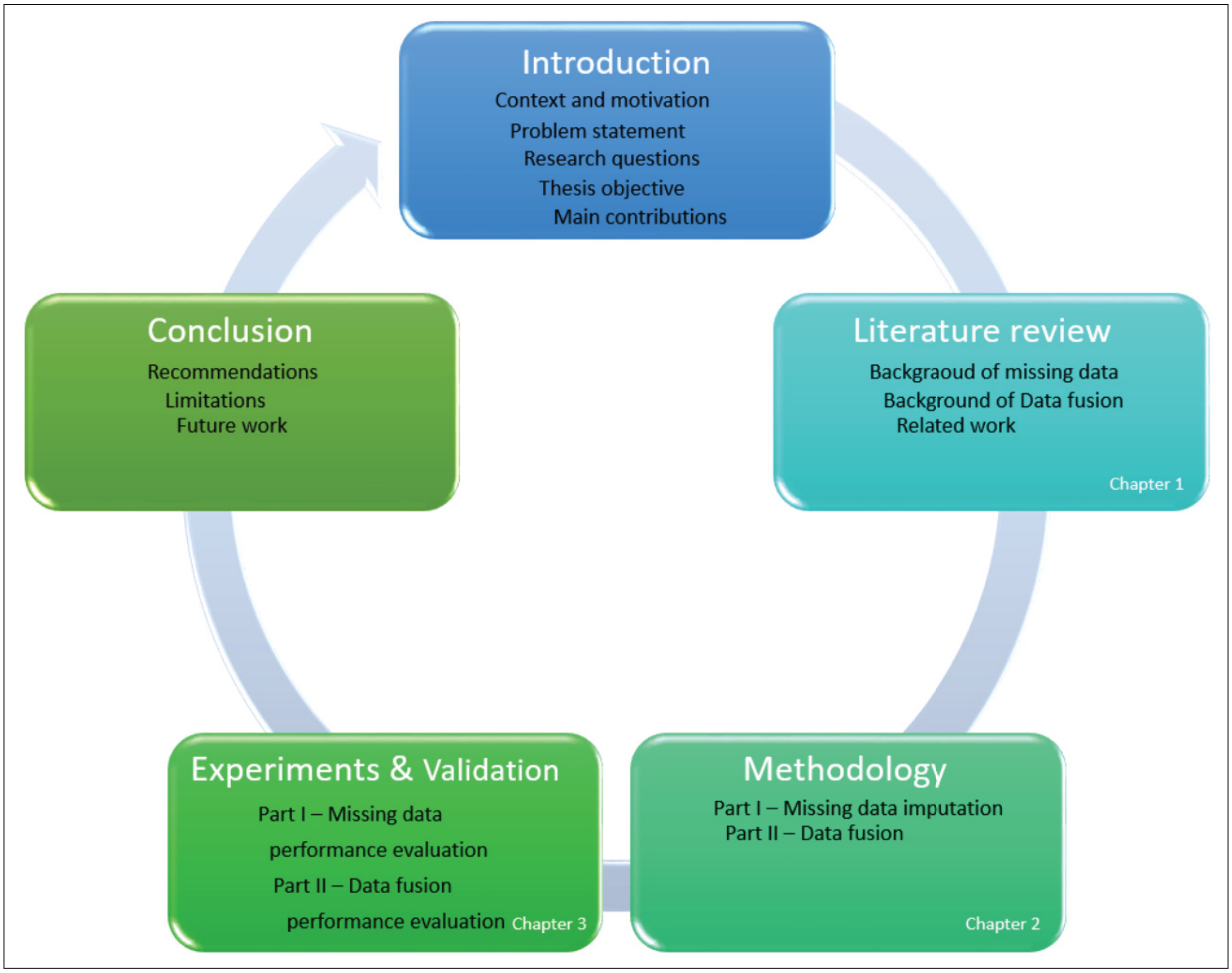


Figure 0.4 Thesis outline diagram

CHAPTER 1

LITERATURE REVIEW

In this chapter, we present a technical background related to the challenges described in the introduction. We divided this chapter into two sections, background and literature review. Firstly we introduce the concepts of missing data and their mechanism and the different data context in which missingness can occur. Then, we present the background of data fusion and its most relevant proposed frameworks. The second section brings the literature review and the state-of-the-art of both missing values and data fusion problems.

1.1 Background of the inconsistency of missing data imputation in datasets

In statistical literature, imputation means "filling in the data". Allan & Wishart (1930) was the first to provide formulas for replacing the value of a single missing observation. Yates (1933) generalized his work to more than one missing observation. The EM algorithm for missing data imputation was introduced in 1977 (Dempster, Laird, and Rubin 1977). Few years later, in 1983, the term "imputation" gained widespread use with the work Panel on Incomplete Data in the overview of the state-of-the-art of imputation technology (Madow, Olkin, and Rubin 1983).

The term "imputation" was firmly established as the mainstream in statistical literature in 1987. Multiple data imputation is now accepted as the best general method to deal with incomplete data in many fields and since that, many other researchers have realized the full generality of the missing data problem. Effectively, missing data has now been transformed into one of the great academic growth industries with up to 500 related scientific publications in the past four years, as can be found in Fig. 1.1. It is a pervasive problem which affects studies quality, producing biased estimates, particularly in analytic and learning processes; such inconsistency can significantly impact the conclusions that are drawn from the data and may bias the decision process, Hatem (2017).

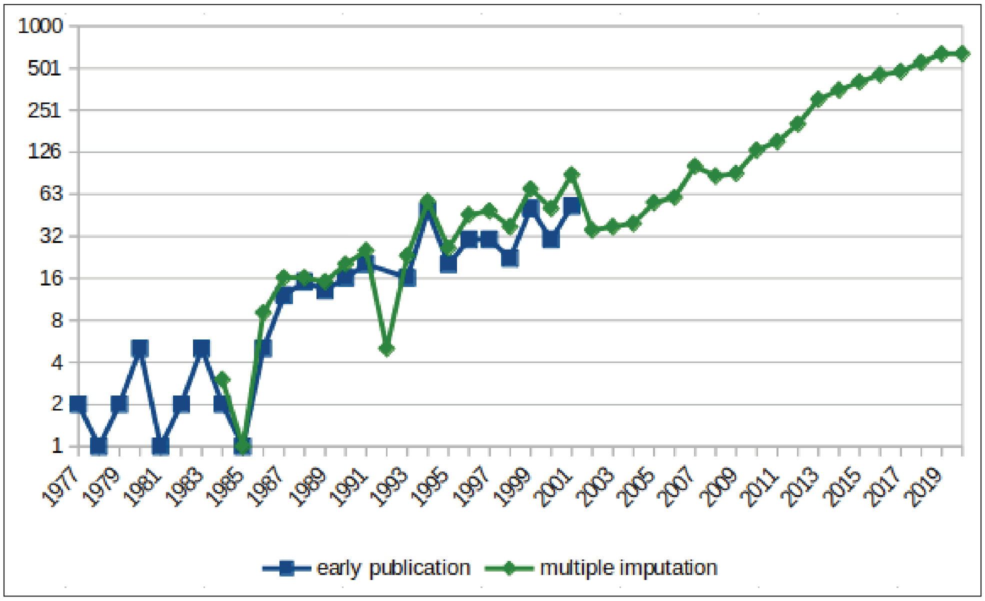


Figure 1.1 The missing values research publication

1.1.1 Missing values problem

Data may be missed intentionally or not. Intentionally can occur, for example, when one turns off a temperature's sensor, causing missing data for the period of time in which it remained off, resulting in a gap for the entire row or unit in a dataset. Another missing data can happen when values are dropped for some specific reason. Then, missing data are intentionally planned in these situations. On the other hand, unintentional missing data are unplanned and its causes are not in perspective or under control, Graham (2012). In this thesis, we define A as an input dataset in a matrix format. From A is derived A' and $A\emptyset$, thus A = (A', $A\emptyset$). Let R be a binary (0, 1) map of A_{ij} with 1 for the datum and 0 for missing, and ψ hold the missing values model parameters. Then the general definition of the missing value model is $Pr(R|A', A\emptyset, \psi)$. In the literature, the adequate missing data treatment is usually in accordance with three of the most prominent mechanism, Schafer & Graham (2002).

1.1.1.1 Missing at random case

For MAR, the probability that an attribute will have missing values depends on another attribute or observation. For instance, a temperature sensor can occasionally fail when the heat exceeds certain limits, or when the sensor dispatch packet but it is not stored for some network failure reason. If this limitation about the sensor operation is known then, MAR is assumed. This case is defined as:

$$Pr(R = 0|A', A\emptyset, \psi) = Pr(R = 0|A', \psi) \tag{1.1}$$

Thus, some internal observed attribute or observation is probable, causing missingness in others.

1.1.1.2 Missing not at random case

In MNAR case, the missing value in an attribute is strictly dependent on the values in other attributes or observations. Continuing with the temperature sensor's example, it may fray over time, producing more missing values as time progresses, and it remains unknown; one individual fails to note this. This case is defined as:

$$Pr(R = 0|A', A\emptyset, \psi) \tag{1.2}$$

Here the probability of being missing depends on another internal observed or unobserved attribute.

1.1.1.3 Missing completely at random

In the MCAR case, all attributes will have the same probability of missing values whereas observed or not. The missing values are not related to the data itself. It implies that the misses are due to some external reason, such as some random interruption during the dispatching or writing process. This case is defined as:

$$Pr(R = 0|A', A\emptyset, \psi) = Pr(R = 0|\psi) \tag{1.3}$$

Therefore, the probability of being missing depends only on ψ parameters. Fig. 1.2, from left to right, shows the defined mechanism for missing values cases.

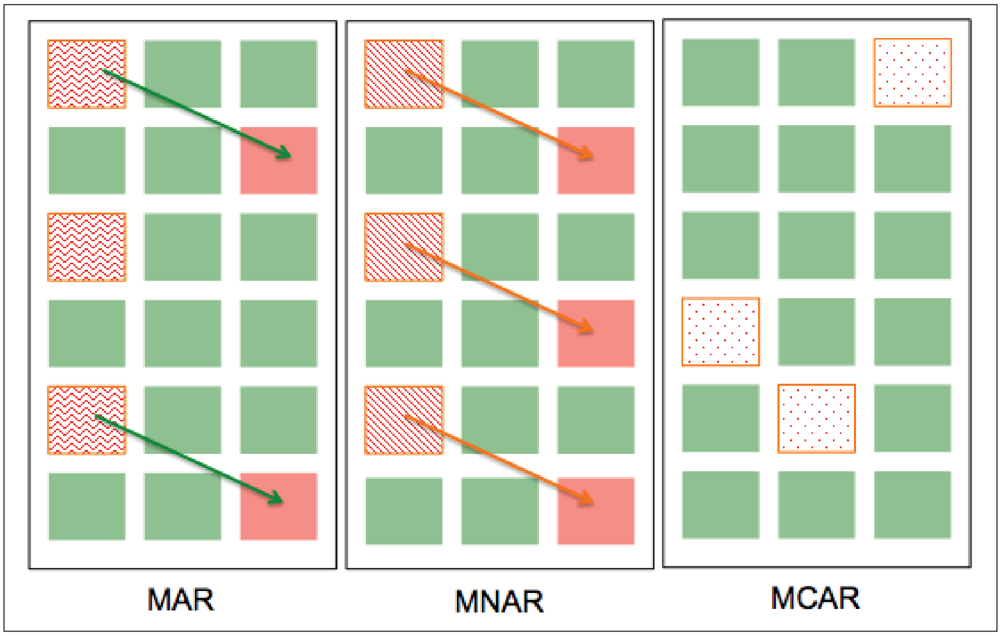


Figure 1.2 Missing values cases mechanism

1.1.2 Dealing with missing values

Over the past years, the missing values problem has been tackled from either a statistical or model-based perspective. While the former was first raised in the earliest half of the last century, the latter was mostly developed in the last two decades and is still evolving leveraged by the machine learning algorithms.

1.1.2.1 Statistical perspective

Solutions from statistical methods such as the mean, median, arbitrary values, end of the distribution, most frequent value (MFV) or mode can be straightforwardly applied and the drawback of such methods is the introduction of some level of bias, Sun & Saenko (2015). For instance, the use of mean or median to impute missing values in continuous attributes can directly affect the attribute distribution. In categorical attributes, according to the cardinality of the attribute, the employment of hot encoders results in a high number of new columns, causing the curse of dimensionality. These problems might bring unsatisfactory and unreliable results in the prediction tasks, Sun, Feng & Saenko (2016).

1.1.2.2 Model-based perspective

Model-based methods are customized machine learning (ML) algorithms that typically map a proposed solution onto a specific problem. An advantage of this approach is its rapid prototyping to meet a specific demand. The disadvantage, however, is the necessity to formulate a solution for every new problem, Bishop (2013). The model-based approach allows univariate and multivariate analyses. Considering an univariate analysis, the values of an attribute are used as a reference to impute its missing ones. In multivariate analysis, one attribute might be useful for inputting the missing values in another one, mainly in the case of a high degree correlation, Petrozziello *et al.* (2018). Nevertheless, a relying issue on such correlation is that even though two attributes have a strong correlation, they can represent different information in a dataset.

1.1.3 Heterogeneous datasets

Another critical demonstration of missing values occurs in scenarios of heterogeneous data Fig. 0.1, which is generated by different sources, usually composed of unstructured, semi-structured, and structured data (e.g., a relational database), Tan *et al.* (2015). It stems from the high variability of data types, usually divided as follows:

- *syntactic*: datasets in which more than one language can be found.
- *conceptual*: differences in modelling the same domain of the information.
- *terminological*: This occurs when the same entity is referred to differently in distinct datasets.

The current sophisticated platforms generate a massive amount of heterogeneous data, especially in IoT, due to the abundant variety of devices, Wang (2017). An example can be seen in the appendix II, Fig. II-1, Fig. II-2, Fig. II-3, respectively. Because of the different data types, continuous, ordinal categorical, and binary, the imputation process's criteria has to be generalized as much as possible to cover all possible kinds, considering that certain type requires different treatment.

1.1.4 Distributed data

In view of big data context, for a given task, the data processing scales horizontally, and the set of machines increases according to the demand. In the last decade, it has been the preferred approach instead run such heavy tasks in a single machine with huge resources, Cordova & Moh (2015), image 1.3 shows how the machine's resources scale works. The dataset is processed in more than one machine, which is fault-tolerant and cheaper. An example of this type of dataset would be the ones larger in size or number of rows. When a certain threshold is reached for a sake of performance, the dataset must be stored and processed in a cluster of machines to take advantage of parallel processing. Yet, to do so, distributed datasets have their own set of complexities, for instance, in Mapuce tasks, Dean & Ghemawat (2008). A proposed solution to overcome the missing data in distributed datasets that usually vary from gigabytes to terabytes is presented in, Petrozziello *et al.* (2018). Their approach is to scale the task into nodes running gradients on a mini-batch process.

1.1.5 Multiple sources

Datasets can be found all over the internet. However, the systematic review method to find multiple data sources for the eligible goal might be time-consuming. An individual should consider which available sources may contain the most useful data, considering that some data sources are more functional than others. Sources of data can be public or nonpublic and determining what to look for and how to obtain the data efficiently is crucial. The information can be, for example, in journals, reports, books, datasets, databases, and so on, Mayo-Wilson, Li, Fusco & Dickersin (2018). It can be structured, semi-structured, and unstructured, and in diverse formats. Because of the inherent complexity of this type of data, the underlying solution

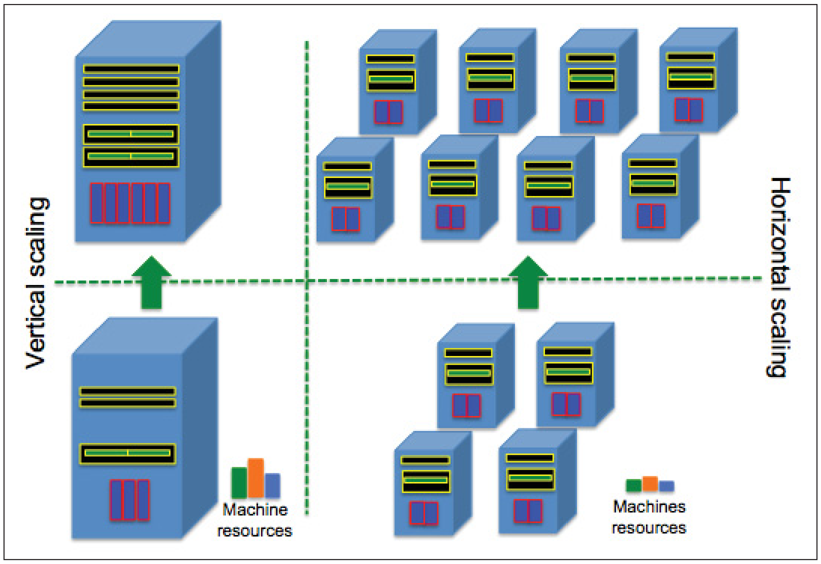


Figure 1.3 Machine scaling - synthetized architecture

for the missing values must consider aspects such as language modeling, word segmentation, and part-of-speech as proposed in, Gudivada, Rao & Raghavan (2015).

1.1.6 Dataset Validation

Data validation is known as an activity verifying whether or not a combination for a single record, column, or larger collection of data is a member of an acceptable set of combinations, Di Zio, Fursova & T. (2016), Schuster & Vitoux (2018). In the relational database management system (RDBMS) and NoSQL non-relation database, the tables have basic functionalities aimed to reduce anomalies and inconsistencies, protecting the data integrity. Specifically in the RDBMS, through the tables connections or other database schemas, it may be useful sometimes to tackle the missing data problem, Geerts, Mecca, Papotti & Santoro (2019). Therefore, when stored in some big data framework such as Apache Spark, the internal resilient distributed dataset (RDD) provides resources that can support the missing data imputation task.

1.1.7 Multiple formats

Contrarily RDBMS, in the distributed/isolated datasets (commonly CSV, XML, JSON files), there is no external relationship. Therefore, by dealing with the missingness problem in those distributed/isolated datasets which the relationship cannot fill the missing values, the algorithm has to learn exclusively from the data available in the dataset itself. This problem remains open, and requires even more attention in big data scenario, which involves multiple sources, modalities, distributed and heterogeneous data.

1.1.8 Discussion

As shown in this subsection, missingness is a central problem of data, and data imputation is an important process to overcome the loss and ensure data quality. Since the very beginning of the last century, different authors have been investigating missing values from different perspectives and employing even more sophisticated solutions according to the computational resources available. Currently, due to big data architecture and complexity, a single method is no longer enough to deal with such a problem. This issue has been tackled mostly through machine learning models. Next, we will present the background, the classification, and the highest spread data fusion techniques.

1.2 Background of Data fusion

Certain knowledge or behaviour about a dynamic environment is better measured and reliable when combining data from multiple sources. In the early '80s, in the, Crowley (1984) publication, they have made an analogy of data fusion principles comparing the uses of sensory skills by the animals in wild environments to track, to identify situations, and to fuse imagery from more than one source from robots. In the same decade, Durrant-Whyte (1987) introduced the data fusion when manipulating robotics and perception, integrating data from different sensors. Faugeras, Ayache & Faverjon (1986) have adapted the fusion techniques to the sound domain. The principle of minimum energy (minimum entropy) used in ANN was approached by, Hopfield (1982) and the assumption of minimizing an energy function would measure if a constraint was violated in a surface reconstruction process with complementary data sources in a parallel neural network. Dasarathy (1997) illustrates the data fusion concepts, conceiving the human brain as a multisensory environment able to fuse different signals. The Eyes, ears, nose, tongue, and skin are sensors that receive signals (sight, hearing, smell, taste, and touch) and process them at different levels and combinations. Therefore, data fusion is a way to obtain improved and reliable information through a collective aggregation of data, which would not be possible if the data had been analyzed separately.

1.2.1 Joint Directors of Laboratories classification

The JDL has defined data fusion as a procedure in which data are associated, correlated, and combined, Hall & Llinas (1997). According to, Castanedo (2013), the JDL classification is the most disseminated concept in the data fusion community. The proposed concept comprises five levels of the fusion process, as follows:

- *level 0:* it is the lowest level, fusing raw signals, pixels, and text. This stage aims to prepare useful information for the subsequent levels.
- *level 1:* receives the output from the precedent level and performs the alignments, associations, correlations, and clustering of other group data techniques. The outcome of this stage is structured data.
- *level 2:* performs inferences regarding the associations and correlations higher than level 1. Mainly this level seeks the relevant relationship between clusters of objects or events (the general patterns). The output is the pertinent inferences.
- *level 3:* evaluates the inferences carried from level 2, looking for possible opportunities or threats and predicting possible outcomes.
- *level 4:* this level evaluates the data from levels 0 to 3 to refine the process cycle.

The first contribution of JDL was to provide a framework with standard nomenclature and a foundation to design a fusion system. Table 1.1 summarizes each level's main activities, and Fig. 1.4 depicts the conceptual JDL framework.

Table 1.1 Summary of the level's main activities

FUSION	The integration of information from multiple sources to produce specific and comprehensive unified data about an entity.
ALIGNMENT (level 1)	Processing of sensor measurements to achieve a common time base and a common spatial reference.
ASSOCIATION (level 1)	A process by which the closeness of sensor measurements is completed.
CORRELATION (level 1)	A decision-making process which employs an association technique as a basis for allocating sensor measurements to the fixed or tracked location of an entity.
CORRELATOR-TRACKER (level 1)	A process which generally employs both correlation and fusion component processes to transform sensor measurements into updated states and covariance for entity tracks.
CLASSIFICATION (level 1)	A process where some level of identity of an entity is established, either as a member of a class, a type within a class, or a specific unit within a type.
SITUATION ASSESSMENT (level 2)	A process by which the distribution of fixed and tracked entities are established, associated with environmental, doctrinal, and performance data.
THREAT ASSESSMENT (level 3)	A structured multi-perspective assessment of the distributions of fixed and tracked entities which results in estimates of (e.g.): • expected courses of action; • enemy lethality; • unity composition and deployment; • functional networks (e.g., supply; and • environmental effects.

1.2.2 Dasarathy's classification

Different from JDL, Dasarathy (1997) has raised a fusion model based on the following input and output data types: Data in/Data out, Data in/Feature out, Feature in/Feature out, Feature in/Decision out, and Decision in/Decision out. Fig. 1.5 shown the framework schematically. Each pair of input and its respective output is defined as follows:

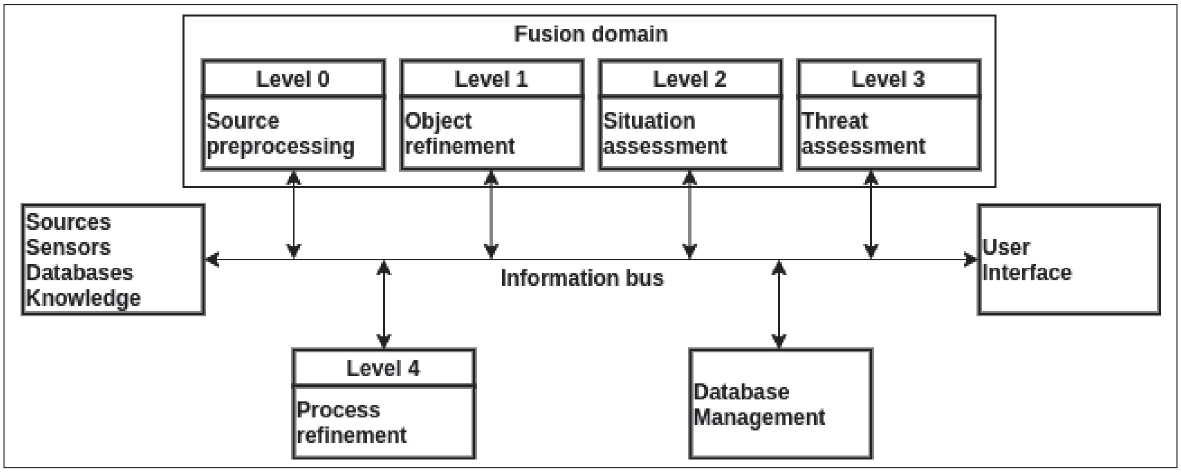


Figure 1.4 Classification based on the type of architecture
Taken from Castanedo (2013)

- *DAI-DAO:* Data In - Data Out, this is the elementary type of data type for fusion. For example, pixels are acquired from an online streaming process or a sensor's signal.
- *DAI-FEO:* Data In - Feature Out, in this hierarchy, the inputted data from a different source is fused, and the output represents some feature. We can mention data from sensors capturing different dimensions can be fused, resulting in a depth feature.
- *FEI-FEO:* Feature In - Feature Out, in this stage, instead of some sensed measure from a sensor, the input is a quantitative or qualitative feature from the system. For example, a shape sensor and a speed sensor where the features can be combined resulting in volume and velocity, a classical fusion task.
- *FEI-DEO:* Feature In - Decision Out, this fusion paradigms receives features, and the output is a decision. For example, according to the input features, the output might be a decision to track or not depending on a priori knowledge.
- *DEO-DEO:* Decision In - Decision Out, this is the last step of fusion. It applies in the cases where the previous fusion stage was performed. For instance, at the moment that a system fuses information from an array of sensors based on prior knowledge and feeds other systems in charge, it takes some action.

The level or schema employed in the fusion process depends on which data, feature, or decision an individual has in hands. Eventually, more than one level must be associated to achieve the primary goal, Pavlidis *et al.* (2002).

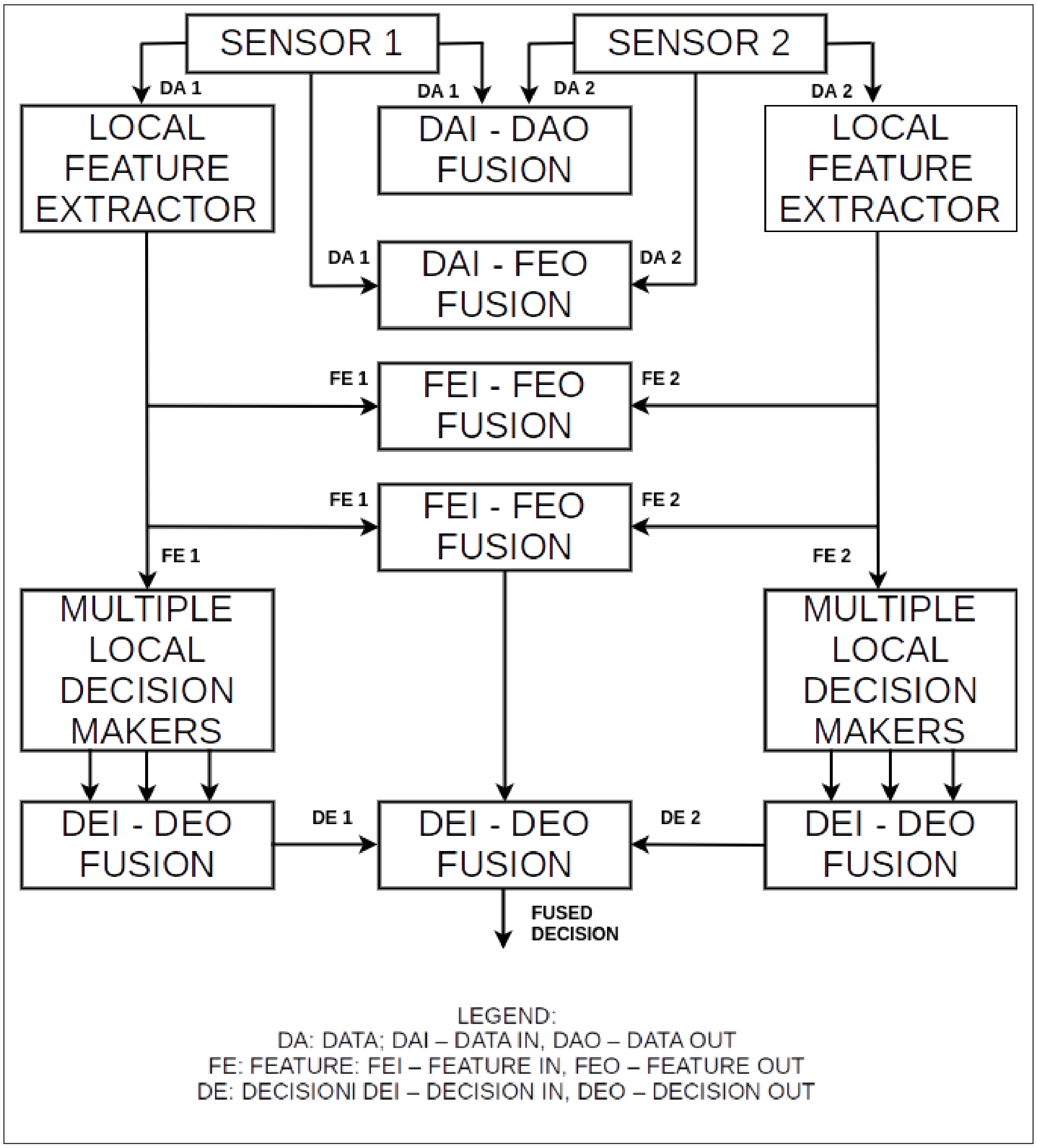


Figure 1.5 Classification based on input and output data type
Taken from Dasarathy (1997)

1.2.3 Fusion stage

Besides the previous perspective, data fusion's new techniques present three different fusion approaches: early, intermediate and late integration. In particular, these techniques describe how the datasets are allocated in a block of matrices structure and how the machine learning algorithm addresses each different one of them. It differs from the JDL and Dasarathy's classification, considering that each integration stage is plugged into a different machine learning arrangement.

1.2.3.1 Early

In early integration, all datasets are merged in a single matrix, and then a single machine learning model analyzes the unified content. The advantage is its fast implementation when compared to more complex approaches. On the other hand, the data structure is disregarded and, consequently, the information underlying behind such structure.

1.2.3.2 Intermediate

Neither early nor late integration deals with the intrinsic relationship among the datasets, the intermediate integration can tackle multiple datasets, absorbing their structure in a single predictive model. It is the most recent data fusion approach and explicitly addresses the multiplicity of data. This integration performs a single joint model able to effectively adds information to the model, Pavlidis *et al.* (2002).

1.2.3.3 Late

In late integration, there is an associated machine learning model to each dataset to fuses its predictions. These three approaches is depicted in Fig. 1.6.

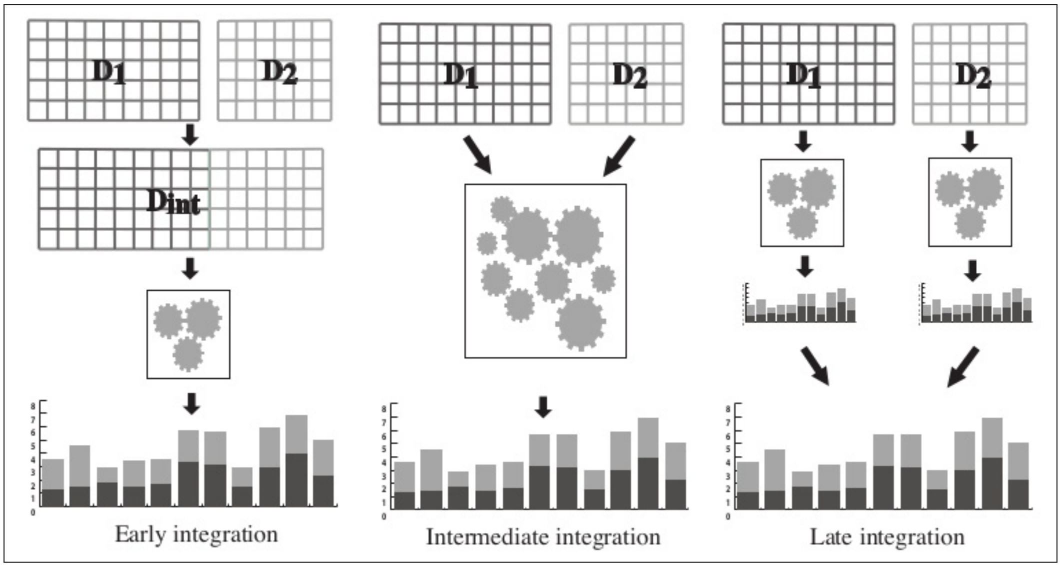


Figure 1.6 Early, Intermediate and Late integration
Taken from Pavlidis *et al.* (2002)

1.2.4 Discussion

In this subsection, we introduced the most accepted data fusion frameworks. Hardly the foundations of data fusion are new. Indeed, through cross-domain data, the authors' proposals attempt to mimic humans and animals to develop awareness about the environment. We discussed the three variants of the most accepted fusion approaches, which are: First, according to different levels of fusion. Second, according to the input and output that could be data, features, or decisions. Third, the combination of the structure in which the data is placed, and the respective machine learning process gives rise to early, intermediate, or late fusion. Next, we will present the related work and state-of-the-art for the two related problems, missing data imputation and data fusion.

1.3 Related work

At this point, we present a review of the state-of-the-art related to the challenges previously discussed. We have divided this section into two parts, missing values imputation, and data fusion, respectively. In the first piece, we present a gamut of a different perspective in which recent researches are addressing the missing values problem. The focus is on the statistical and model-based approaches. The second part is dedicated to data fusion and covers the most important techniques and frameworks proposed in this field.

1.3.1 Statistical techniques used to overcome missing values problems

Often, practitioners and researchers face the problem of missing values when treating either single or multiple datasets under ML, DM, or data fusion processes. Although many statistical methods can impute missing values, they can lead to undesirable outcomes, such as a) distortion of the original variance, b) distortion of covariance and correlation with other columns within the dataset, and c) overrepresentation of the most frequent value (MFV) when a large number of observations are missing, Schafer & Graham (2002). Moreover, a simple deletion can considerably reduce the data for any multivariate analysis. It may be even worse if the loss affects a variable of interest of a small dataset. Different statistical methods make different assumptions and have both advantages and disadvantages.

In, Little Roderick & Rubin (2002), the authors have defined the most common statistical approach to missing values imputation according to the mechanism of missing values:

- *Mean and median:* The assumptions of using this method are in the MCAR mechanism's presence and depend on its distribution. If it is Gaussian, the mean is suggested; otherwise, the median is preferable.
- *Most frequent value:* The uses of this method are in the MAR mechanism's presence. This technique is analogous to mode value, mainly used for categorical values.
- *End of tail:* The application of this technique is in the MNAR, and it implies assigning rareness values of the population to the missing values.

For, Wielenga (2007), a common point among different statistical approaches is its implementation; The advantage of such an approach is that it is usually easier and faster to reconstruct and obtain a complete dataset. However, it can distort the original distribution of the attribute. In Table 1.2, we summarize the statistical methods, their application, and limitations.

Table 1.2 Statistical approaches for missing values imputation

Imputation Method	**Assumption**	**Application**	**Limitations**
Mean Median	MCAR case	• Straightforward to implement • Fastly obtains complete datasets	• Original variance covariance and correlation distortion
Most frequent value (MFV)	MAR case	• Straightforward to implement • Fastly obtains complete datasets	• Distortion the relation of the MFV with other dataset's variables • May lead to an over MFV representation
End of tail	MNAR case	• Straightforward to implement • Captures missingness if there is any	• Distortion the original distribution
Arbitrary value	MNAR case	• Straightforward to implement • Captures missingness if there is any	• Hard to decide which arbitrary value to use

1.3.2 Model-based techniques used to overcome missing values problems

Over the last decade, machine learning has been applied in widespread practical problems. Aside from the statistics approach, the research community has used machine learning algorithms to address the missing value problems, Bishop (2013). Practitioners must choose a suitable machine learning technique to cover different aspects of the missing data, such as data heterogeneity, data format, big data environment, missing mechanism, and so on. Although the machine learning algorithm presents some flexibility, they commonly need to be re-written or have their parameters adjusted to meet the datasets' requirements that constantly change. For this reason, in the last few years, we have witnessed a significant increase in model-based solutions to properly address the problem of missingness when big data is considered, Wang (2017). Bellow, we present and discuss the frameworks that currently represent the state of the art.

1.3.2.1 Generalized low rank model

Udell, Horn, Zadeh & Boyd (2016), researches in model-based, proposed methods such as Principal Component Analysis (PCA) in the GLRM framework. Their approach compresses the dataset with continuous, discrete, and binary values, into a low dimensional vector space. The compressed dataset is composed by features that explain a maximal amount of variance, and the value of these features is used to impute missing values. The framework adopted some extensions of PCA, for example quadratically PCA, alternating minimization PCA, to perform on different matrix structures, such as sparsity or nonnegative. In fact, this model approximates the dataset in two low dimensional factors by minimizing an objective function. The drawback is that the data must correspond to the ML model assumptions. The GLRM uses quadratically regularized PCA, a model with normal distribution, varying between 0 and 1. The research highlighted the framework limitation as not performing in non-normal distribution.

1.3.2.2 Distributed Neural Network

The authors in, Petrozziello *et al.* (2018) presented a DNN, which deals with datasets composed of continuous values using mini-batch stochastic gradient descent. Their framework runs over a big data platform and uses a data-parallelization schema with synchronization and a central coordinator. This approach is only possible when built-in modern systems can scale vertically or horizontally, where workers mean parallel processes operating in different machines, as illustrated in Fig. 1.7. The restriction is related to zero attribute values. Their model performs better with continuous values while struggling with the zero's ones.

1.3.2.3 Fuzzy similarity

The fuzzy similarity is used by, Baraldi, Maio, Genini & Zio (2015) in a multidimensional time series context of on-line conditions, in which the missing values are reconstructed with the average of the reference data. Their method comprises three steps. First, the segment of the series that contain missing is confronted with the completed ones. Then, the following step

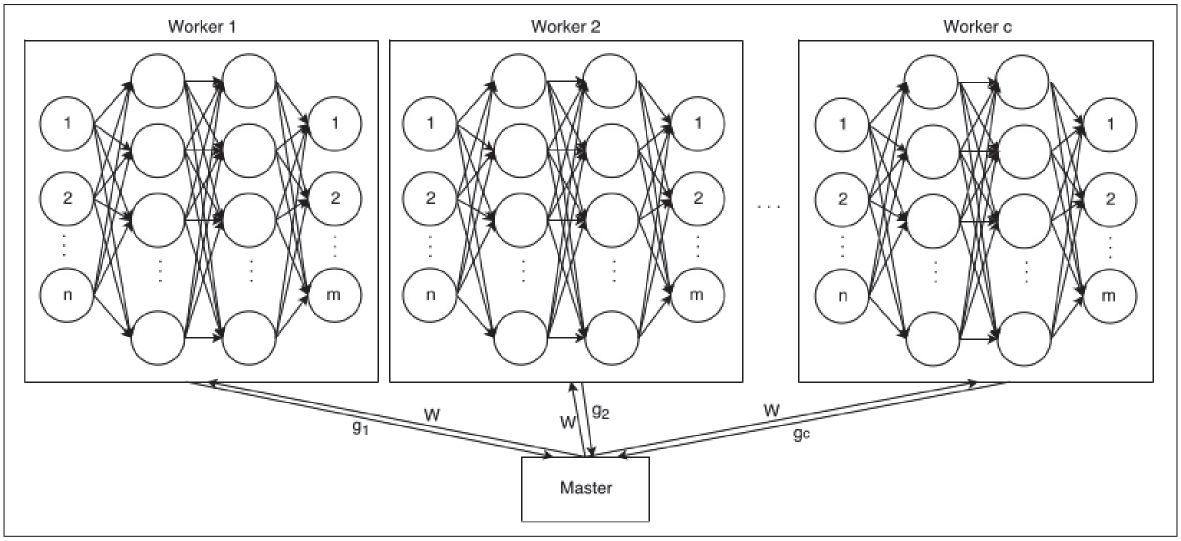


Figure 1.7 Distributed Neural Network architecture
Taken from Petrozziello *et al.* (2018)

aims to weigh the complete segment. The missing values are filled with the weighted segment's average from the previous step in the third step. The limitations here are that the reconstruction of the missing values based on the weighted means of the data in the training task leads to biases and the fact that the approach is restricted to time series data.

1.3.2.4 Conceptual reconstruction

Aggarwal & Srinivasan (2001) offered a conceptual reconstruction model based on PCA with continuous values in a context in which a large percentage of data is missing. They express the data in terms of salient directions determined by the most prominent variance as depicted in Fig. 1.8. They use the structure's correlation of the data to produce concepts that mimic the original dimensions. In the imputation procedure, the concepts work as a reference to fill the missing values. The model's weakness derives from the fact that it depends of a dataset with high correlation among attributes.

In Table 1.3, we summarize the objective, contributions and limitations of the model-based related work.

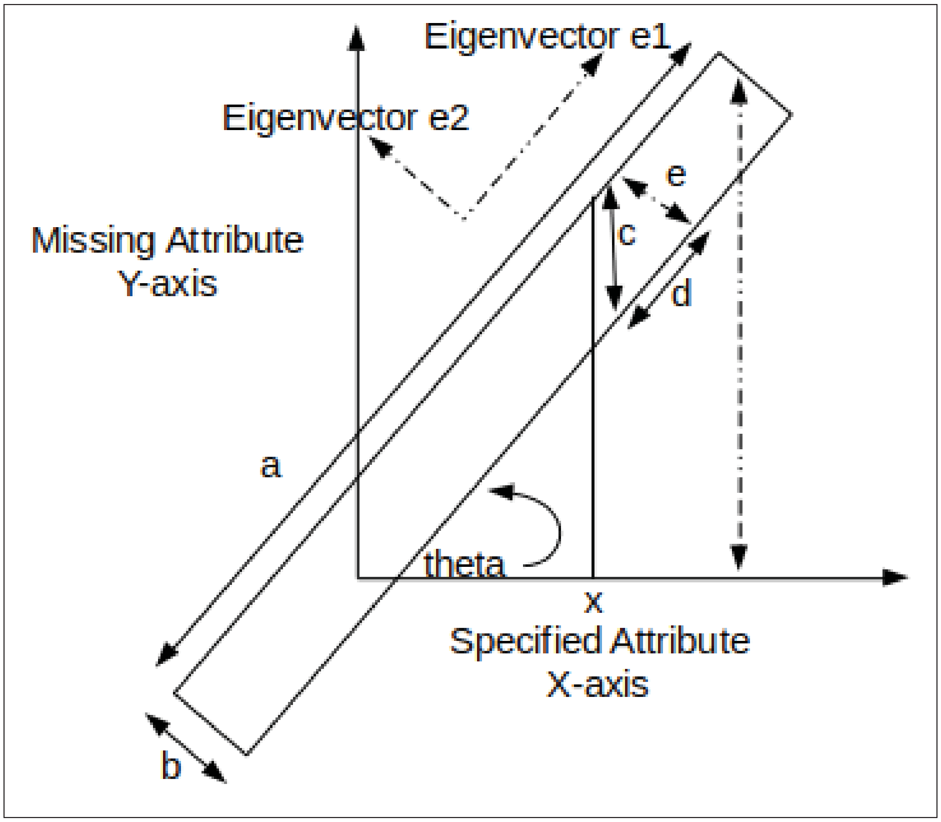


Figure 1.8 Predictability for a simple distribution
Taken from Aggarwal & Srinivasan (2001)

1.3.3 Discussion

Comparing the state-of-the-art methods (besides the statistical techniques), we found four model-based approaches for missing value imputation: GLRM, DNN, Fuzzy Similarity, and Conceptual Reconstruction. In general, the significant limitations highlighted in the related work are:

- data distribution assumptions.
- deficiencies in dealing with heterogeneous data types.
- the need for attributes correlation.
- the assumption between the ML model and data type.
- i.i.d, which is rare in real-world datasets.

Table 1.3 Related work for based-model missing values imputation comparison

Related work	Objective	Contributions	Limitations
GLRM Udell et al. (2016)	Consider the generality of missing values problem	• Imputation of missing values across different data types • Handle the full gamut of PCA and regularizers	• Assumes normal distribution of the data • Categorical values need to be encoded
DNN Petrozziello et al. (2018)	Implement a Distributed Neural Network for imputation	• Imputation of missing values in the big data context	• Struggles with attributes which value is zero • Works only with continuous values
Fuzzy Similarity Baraldi et al. (2015)	Imputation missing values in a time series dataset	• Uses Fuzzy similarity method applied in on-line conditions	• Weight the data's mean driven to bias • Works only with time series data
Conceptual Reconstruction Aggarwal et al. (2001)	Imputation missing values when it is massive in a dataset	• Exploit the dataset attributes correlation	• Depends on attributes correlation • Supports only continuous values

The overall related work comparison is summarized in Table 1.3. Our proposal to surpass such restrictions will be exposed at the conclusion of this chapter.

1.3.4 Data fusion techniques

Frequently practitioners and researchers face data fusion challenges when treating multiple datasets using ML or DM processes, Housfater, Zhang & Zhou (2006). Indeed, many fusion methods can be employed according to the goal and the strategy adopted to overcome the problem.

1.3.4.1 Statistical fusion

In, de Oliveira & Kedem (2017), presented a statistical approach to deal with the semiparametric inference over the probability distribution regarding two or more different samples of data. It is based on a scenario of distributions that allows formulations of hypothesis tests to validate the fused model in terms of variance, covariance, and correlation between the datasets, de Oliveira & Kedem (2018). The work limitations are: first, the need for a reference distribution to perform inference fusion. Second, it struggles in non-normal distribution, the case of most real-world datasets. Thus, this technique works only in limited cases and not advisable for a distributed scenario.

1.3.4.2 Gaussian processes

Aside from semiparametric inferences, research on DF proposes methods such as Gaussian process (GP), a non-parametric Bayesian learning technique. The kernel-based approach is common, derived from the non-stationary kernel, normally applied for multi-task problems with dependent processes that incorporate uncertainty and incompleteness data. Although, as asserted in, Vasudevan, Ramos, Nettleton & Durrant-Whyte (2011), the integration of datasets is done one at time, so they are not able to know the fusion structure process of the whole model. The drawback is that the fusion covers a unique data type and modality; only sensor data is fused. The scope of work is limited in terms of data variety.

1.3.4.3 Apriori methods

In, Botega, Junior, Pereira, de Oliveira, Saran, Ladeira & Isotani (2019), they use a framework based on the Apriori algorithm to classify words in an unsupervised NLP process. It supports the analysis for public security services, enriching situational awareness and enabling better employment of resources. Also, it provides more accurate and fast responses to every call. The restriction is related to a process that covers a unique data type. Another drawback is the high human interactions' demands. So, it is not a multi-purpose framework but a customized application.

1.3.4.4 Matrix factorization

Besides the statistics approach, a state-of-the-art matrix factorization named data fusion by matrix factorization (DFMF) framework has been developed by, Zitnik & Zupan (2015). The DFMF tackles DF processes in heterogeneous datasets to fuse feature-based representations. They propose a new method for ontologies, associations, and networks, considering the relationship among datasets and integrating all feasible data, whether it is connected or not. Still, they have restricted the fused datasets towards the observation of the interest upon domain specialist consultations. Wang *et al.* (2019) proposed the SelMFDF, a framework built over the DFMF that performs on multi-relational datasets, and is also able to weigh the inter-relation and intra-relation between the related datasets.

In both DFMF, Zitnik & Zupan (2015), and SelMFDF, Wang *et al.* (2019) there is an important disadvantage. The fusion process is performed constraining the datasets to obtain symmetric dimensionalities, setting up the low-rank parameter for optimization in the matrix factorization. The experiments were performed only in a supervised scenario that demands labeled datasets, which is not the case for most available datasets. The overall fusion methods comparison is summarized in Table 1.4.

1.3.5 Discussion

In comparison with state-of-the-art methods, we have found four data fusion approaches: Statistical fusion, Gaussian processes, Apriori, and Matrix factorization. The significant limitations highlighted in the related work are:

- data distribution assumptions and data variety in Statistical fusion.
- deficiencies in dealing with heterogeneous data types, for example, in the Gaussian process.
- high human interactions' demands in Apriori algorithms.

Table 1.4 Related work data fusion comparison

Related work	**Objective**	**Contributions**	**Limitations**
Statistical fusion Benjamin et al. (2018)	To fuse statistical information from multiple sources	• Develop a Bayesian analysis of the density ratio model focusing on estimating the distributions of different data sources	• Need of a reference distribution to perform the inference fusion • Struggles in the presence of non-normal distribution
Gaussian processes Vasudevan et al. (2011)	To demonstrate the use of the multi-output dependent Gaussian processes	• Derivation and use of non-stationary kernels for multi-task problems	• The fusion covers a unique data type and modality • Only sensor data is fused
Apriori Botega et al. (2019)	To improve the situation of awareness in the human operations systems	• Deals with fusion of semantic information quality criteria	• The fusion covers a unique data type • High human interactions are demanded
Matrix factorization Zitnik et al. (2015)	To deal with collections of heterogeneous datasets	• Proposed a new Matrix factorization fusion algorithm	• Constrains the datasets to obtain symmetric dimensionalities • Performs only in supervised scenario

- in matrix factorization work, the restrictions are: the constraining datasets dimensionality, and it performs only in labelled datasets.

These are the limitations found in the state-of-the-art data fusion framework. Considering such restrictions we propose a data fusion framework presented in the conclusion of this chapter.

1.4 Conclusion

In section 1.3.3, we discussed the restrictions found in the statistical and model-based approaches. To tackle the majority of limitations in section 1.3.3, related to the missing values problem, we propose a framework to tackle continuous, discrete, categorical, and binary data in the same dataset without an extra transformation. The framework performs in linear and nonlinear datasets and is not dependent on the datasets distributions. Thus, such framework can be employed in a broader scenario, configured in a single robust solution for missing values problems in a wide variety of datasets. Based on these objectives, we also propose an innovative stacked ensemble framework to impute missing values based on data topology, Bubenik (2012), composed of three steps: adapted RF, the Jaccard index, and Bayes probability. In Table 1.5 it is shown a summary comparison of our proposed framework. In Fig. 1.9, we depict the gap intersection that it covers.

Table 1.5 Summary of model-based methods coverage for different data types

-	**Continuous**	**Discrete**	**Categorical**	**Binary**
GLRM	✓	✓	✓	✓
PCA	✓	✓	✗	✗
DNN	✓	✗	✗	✗
Fuzzy Similarity	✓	✗	✗	✗
Conceptual Reconstruction	✓	✗	✗	✗
Proposed Framework	✓	✓	✓	✓

To overcome the limitations from one to five related to data fusion problem, discussed in the section 1.3.5, we expose an innovative stack framework based on DL techniques to fuse the heterogeneous datasets. Our project matches a Restricted Boltzmann Machine (RBM) algorithm to perform feature learning and ensure that only significant ones will remain in the model. We use Natural Language Process (NLP) for the dataset that contains the text data type, and a block-based matrix tri-factorization (MF) algorithm to deal with the fusion process. The overall comparison in terms of fusion methods is summarized in Table 1.6. Fig.1.10 illustrates the gap in the literature review that our proposed framework covers.

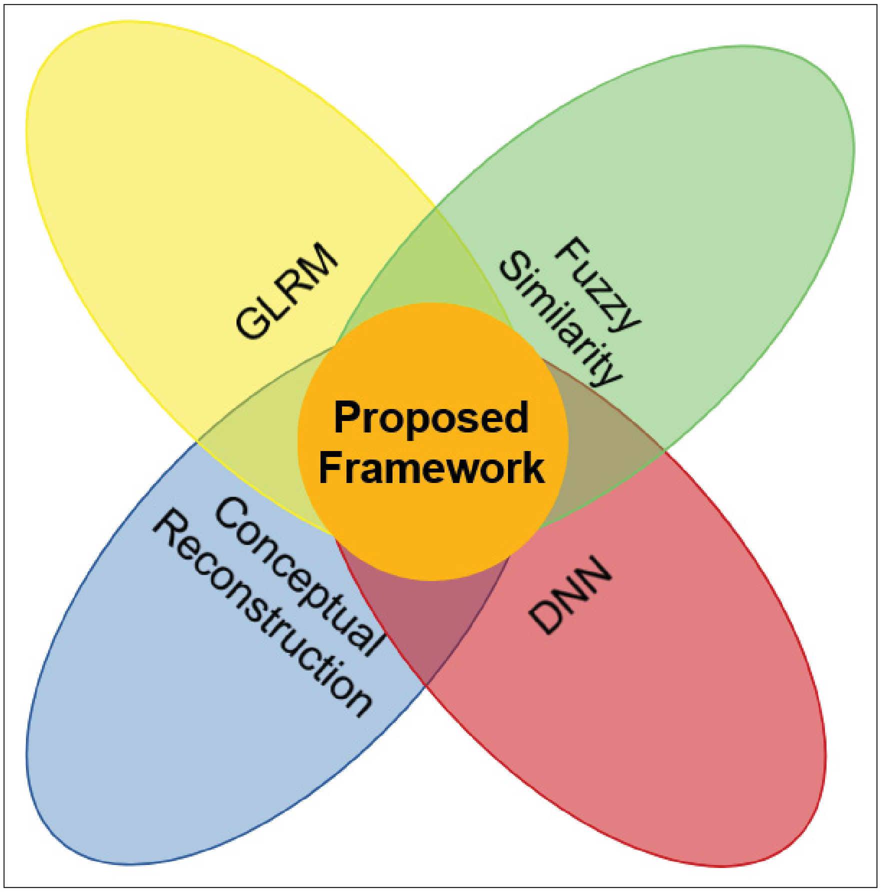


Figure 1.9 Gap found in literature review addressed by our proposed missing values imputation framework

Table 1.6 Summary of fusion methods and its integration level

-	**Kernel based**	**Integration Level**
Statistical fusion	✓	Early
Gaussian processes	✗	Intermediate
Apriori	✗	Early
Matrix factorization	✓	Late
Proposed Framework	✓	Intermediate

The intersection of the related work problems has driven us to propose two versatile and comprehensive frameworks able to scale and perform in a big data context. These are the

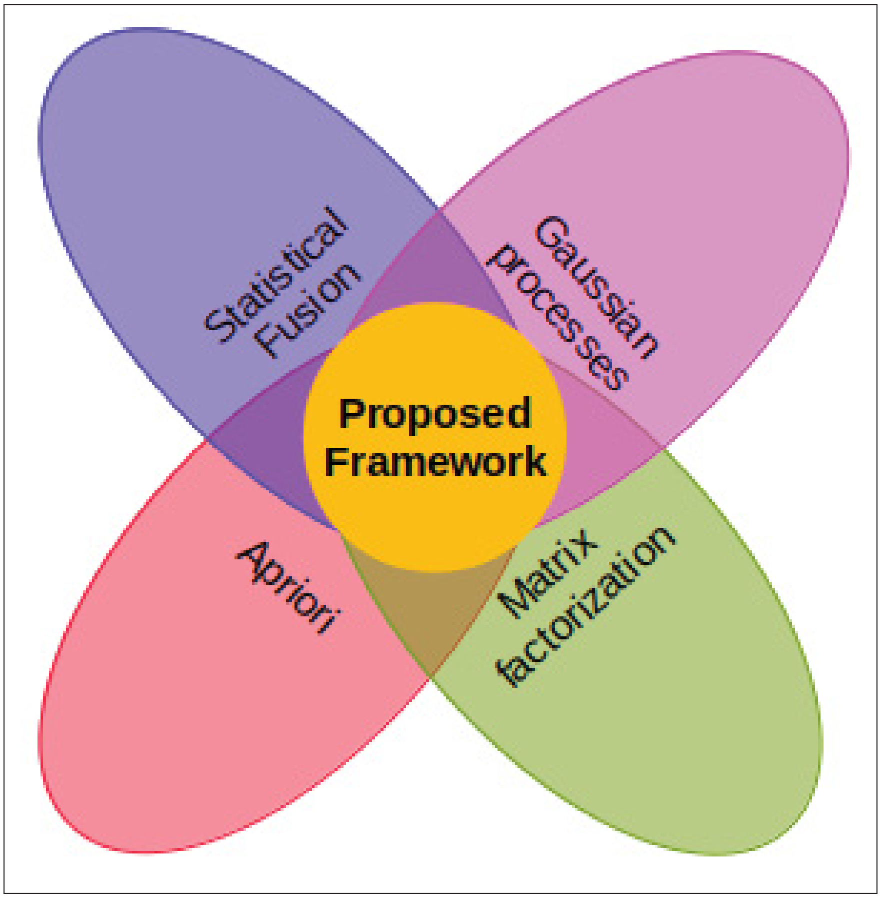


Figure 1.10 Gap found in literature review covered by our proposed data fusion framework

premises behind the missing value imputation and data fusion frameworks that we will present in detail in the next chapter.

CHAPTER 2

METHODOLOGY

This chapter introduces our methodologies proposed to overcome the problems raised in section 0.2 and address the research questions in section 0.3. Then, we demonstrate our algorithms, discussions parts I and II, and finally, the conclusion.

2.1 Part I - Missing values imputation

Our approach provides an extensible framework capable of processing heterogeneous raw data, addressing thereby, the problem statement highlighted in section 0.2. To do so, our framework should avoid: first, making assumptions between data type versus ML model (e.g., the distribution of the data might be an assumption for some ML model), second, produce a narrowed solution that works in a specific data type as those previously discussed in related work, section 1.3. According to, Bengio, Courville & Vincent (2012), the gain in this representation could vary considering its distribution or sparsity (e.g., a node in a decision tree, or one of the units in a restricted Boltzmann machine). A distributed representation of the data means that a large number of its possible subsets can be useful in representing the data in its totality.

2.1.1 Proposed missing values imputation framework

2.1.1.1 Ensemble learning for heterogeneous missing data imputation

We want to reuse different data samples without missing values to reconstruct the portion that suffers from that missing. Towards this end, we propose an innovative stacked ensemble framework to infer missing values based on data topology, Bubenik (2012), comprised of three phases: adapted random forest, the Jaccard index, and Bayes probability.

2.1.1.2 Data ingestion

The preprocessing phase is responsible for data ingestion and differentiating the rows with the complete information from those with missing values. In the given dataset *A*, in an *m* x *n* matrix format, *m* and *n* stand for the row vector and attribute, respectively. As depicted in Fig. 2.1, the entire dataset *A* consists of two portions of rows data: rows without missing values (complete rows) and rows that contain missing values.

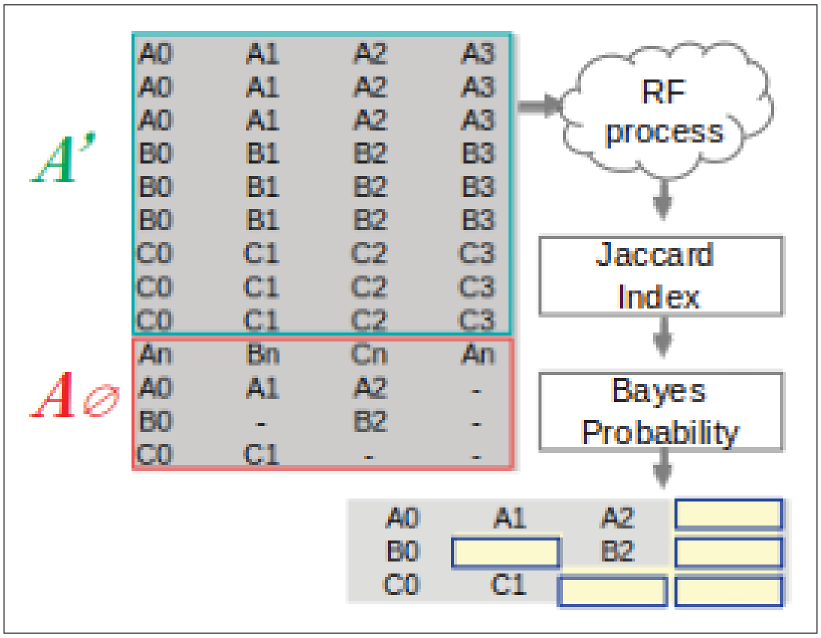


Figure 2.1 Synthetized data ingestion mechanism of ensemble learning for heterogeneous missing data imputation taken from Carvalho *et al.* (2020)

The dataset *A* is subdivided into two parts, generating *A'* and *A*∅. *A'* is a subset of the dataset *A* without missing values, and *A*∅ is a subset of the dataset *A* that contains the rows with the missing values. After spliting the dataset *A* into *A'* and *A*∅, both subsets are inputs of the ensemble adapted RF method.

2.1.1.3 Adapted Random Forest

Thanks to the possibility of combining multiple decision trees, we use the ensemble RF algorithm, which performs better than an individual model, which in turn addresses **research question 1**. Considering the variety of data types mentioned above, we exploit an ensemble ML framework. Ensemble methods train multiple learners and then combine them, known as a state-of-the-art learning approach. It is notable that an ensemble is usually more accurate than a single learner, and ensemble methods have already achieved great success in many real-world tasks, Zhou (2012). The RF produces better results because the trees are formed by different Bootstrap Aggregation. Before presenting the final algorithm for the RF adaptation we present below the steps for training.

2.1.1.4 Bootstrap Aggregation (Bagging)

Bagging is the mechanism adopted to reduce the trees' variance as well as the correlation between the attributes, resulting in multiples trees with different variances (2.1). We used replacement of the row vectors a_m and sampling different dataset attributes a_n, as follows:

$$nattr = \log_2 a_n + 1 \tag{2.1}$$

We have made two important adaptations in our RF. The first was to carry on heterogeneous data types in the same dataset. The second was to store the data subset related to each node of the RF, according to the iterative top-down construction. Combining these two adaptations in the RF with the Jaccard index and Bayes probability, it allowed us to achieve better results when compared with the benchmark (Statistical methods and GLRM framework) for heterogeneous datasets.

2.1.1.5 Gini index

We use Gini Index (2.2) to weight the proportion of the classes of each attribute and select the one with the smallest impurity, defined as:

$$\text{Gini}_i = 1 - \sum_{k=1}^{n} p_{i,k}^2 \tag{2.2}$$

The value p refers to the proportionality of a_{mn} in the attribute a_n developing the learning phase.

2.1.1.6 Jaccard index

In the second phase, following the RF process, we inject the $A\emptyset$ portion of the data that contains missing values and greedily measures its similarity, using the Jaccard index (2.3), among each row with the missing values $a_m\emptyset$ and a_m rows of the subsets stored previously of each node for all trees in the RF process.

$$\text{J(X,Y)} = \frac{\left|a_m \cap a_m\emptyset\right|}{\left|a_m \cup a_m\emptyset\right|} \tag{2.3}$$

The set of rows a_m with the maximum similarity to the row $a_m\emptyset$ is computed. This step will retain the set of rows a_m that have the maximum similarity to the row $a_m\emptyset$. The goal of this step is to filter the set of the rows in the dataset without missing values that are most similar to the row that has missing values.

2.1.1.7 Bayes probability

In the final step, the imputation in the missing values is applied based on Naive Bayes probability (2.4), and the value of the attribute a_{mn} is assigned to the attribute $a_{mn}\emptyset$. This phase aims to define the value with the maximum probability to imputing the missing values.

$$\text{P}(a_{mn}\emptyset\middle|a_{mn}) = \frac{P(a_{mn}\emptyset\middle|a_{mn}) * P(a_{mn})}{P(a_{mn}\emptyset)} \tag{2.4}$$

2.1.2 Proposed missing values algorithm

We present in detail the algorithm of our proposed framework and discuss the typical complexity of working with heterogeneous data types in the same dataset. Regarding categorical values, in real-world datasets, it is normal to find a large number of categorical columns. When these contain high cardinality (many different values), the traditional hot encoders found in the features engineering literature (e.g., one-hot encoding, count or frequency encoding) lead to a high number of new columns. That might affect the model due to the curse of dimensionality or a considerable loss of original information, Schafer & Graham (2002).

Regarding numeric values, it is often necessary in the presence of numeric values to employ some feature transformation strategy, e.g., normalization, scaling, or binning, according to the numeric type (continuous or ordinal), although it may affect the ML results. Thus, special attention is required in the preprocessing phase of the ML pipeline. All notations and parameters used in the missing data imputation are summarized in Table 2.1.

Table 2.1 The notations and parameters in the paper are summarized below

Input notation	**Description**
A	Input dataset in a matrix format
A’	Training set with completeness values
$A\emptyset$	Test set with missing values
a_m	Row vector of A’
a_n	Attribute of A’
a_{mn}	Datum of a_m
a_m $\emptyset$	Row vector of $A\emptyset$
a_n $\emptyset$	Attribute $A\emptyset$
$a_{mn}\emptyset$	Datum of $a_m\emptyset$
nfld	Number of folds cross validation
nattr	Number of attributes in the bagging
msiz	Minimum decision tree size
tst	Type of dataset *(un/labelled)*
ndt	Number of trees
mdpt	Maximum decision tree depth

Algorithm 2.1 Ensemble learning for heterogeneous missing imputation

```
   Input: A', A∅, nfld, mdpt, msiz, tst, ndt
 2 for ndt_i in [ ndt ] do
 4     $j \leftarrow 0$;
 6     while $j <= nfld$ do
 8         $Anfld \leftarrow$ split A' in nfld folds;
10         compute random forest (Anfld, mdpt, msiz, tst, ndt_i) ;
12         update root trees ← random forest results
14         if tst == supervised then
16             compute accuracy ← metrics ( root trees, tst );
17         end if
18     end while
19 end for
21 while tst do
23     $\vec{v}$ = call fit ( tst, root trees ) ;
25     sim ← 1;
27     while $\vec{v}$ do
29         if Jaccard Index($\vec{v}$, tst) <sim then
31             compute sim = $\vec{v}$;
32         end if
33     end while
34 end while
36 if $sim > 1$ then
38     return argmax(sim)
39 else
41     return sim
42 end if
```

A summary of the framework reviewed in 2.1 and its inputs is presented in Table 2.1 and the main steps of algorithm 2.1 are as the following sequence:

- the subset *A'*, without missing values, and the subset *A*∅ containing missing values are inputs of the framework.
- subsequently, the subset *A'* is used to learn the RF algorithm. Besides this subset, the algorithm receives the following parameters: *nfld, mdpt, msiz, tst, and ndt*. (line 5), see Table 2.1.
- we adapted the nodes and terminal leaves so they could store the subset of the data. (line 6).

- in the inference phase of the framework, we evaluate the similarity between a_m and $a_m\emptyset$ applying the Jaccard index throughout the resulting trees. The Jaccard index measurement emphasizes the similarity between the subsets without missing values and the rows vector with missing values. (line 13).
- next, all a_m row vectors that satisfy the minimum Jaccard index is computed in line(14).
- if we find a single row vector in the *root trees* with minimum Jaccard index, the value of this attribute will be the reference to impute the attribute $a_{mn}\emptyset$. (line 17).
- otherwise, if we found more than one row in the *root trees* with the minimum Jaccard index, the Bayes probability is employed to define the most probable value to impute the missing data line(16). In Fig. 2.2, we synthesized the mechanism behind the algorithm.

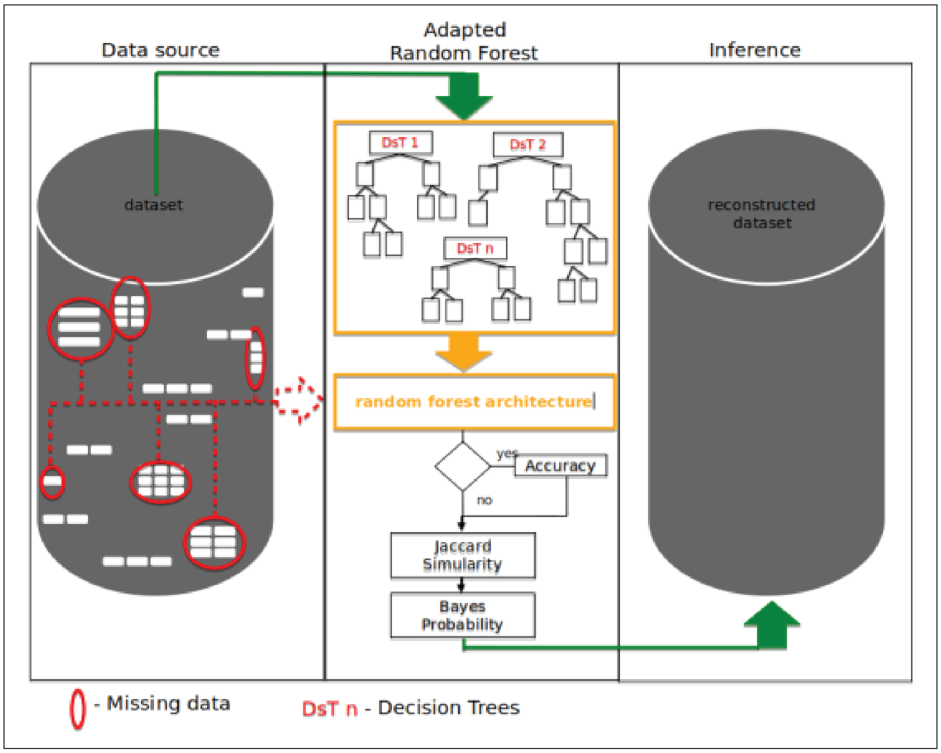


Figure 2.2   Conceptual framework of ensemble learning for heterogeneous missing data imputation taken from Carvalho *et al.* (2020)

2.1.2.1 Discussion

Our proposed framework with the three phases presented above is different from other based-model approaches found in the literature. By applying the ensemble RF step, we have different data representations (trees with different variances) according to the bagging process in the learning phase. The Jaccard index works as a filter, selecting rows without missing values with the maximum similarity to rows that contain missing values. The set of the most similar rows without missing is the input to the next step, Bayes probability. When applying Bayes probability we will have, among all the most similar rows, the attribute with the maximum probability value to impute the missing ones. Thereby, the stacked set of techniques we propose in our framework is responsible to address the **research question 2** and, ensuring such robustness in our framework. Indeed, we are considering the generalization of the of data representation. Thus, with the presented ensemble learning imputation, we can reconstruct a dataset that suffers from missing values while keeping the statistical strength and, at the same time, mitigating the undesirable outcomes, such as distortion of the original variance, distortion of covariance and correlation with other columns within the dataset, overrepresentation, among other problems. All these steps are illustrated in Fig. 2.3.

2.2 Part II - Data fusion

In this section we propose a comprehensive fusion ensemble learning model able to process multiples datasets. Our approach addresses the problem statement highlighted in section 0.2 enabling fusion in both labeled and unlabeled datasets, providing a relaxation in datasets dimensionalities, and also requiring minimal human interaction in the process and working with heterogeneous and multi-domains datasets. By these set of propositions, we can oversome the limitations pointed out in the related work section. According to, Pavlidis *et al.* (2002), the integration based in the abstraction level allows features from different representations to be combined in the same format of representation. We exploit the intermediate integration of data fusion in the proposed framework because the information about the structures of the heterogeneous data may produce better results.

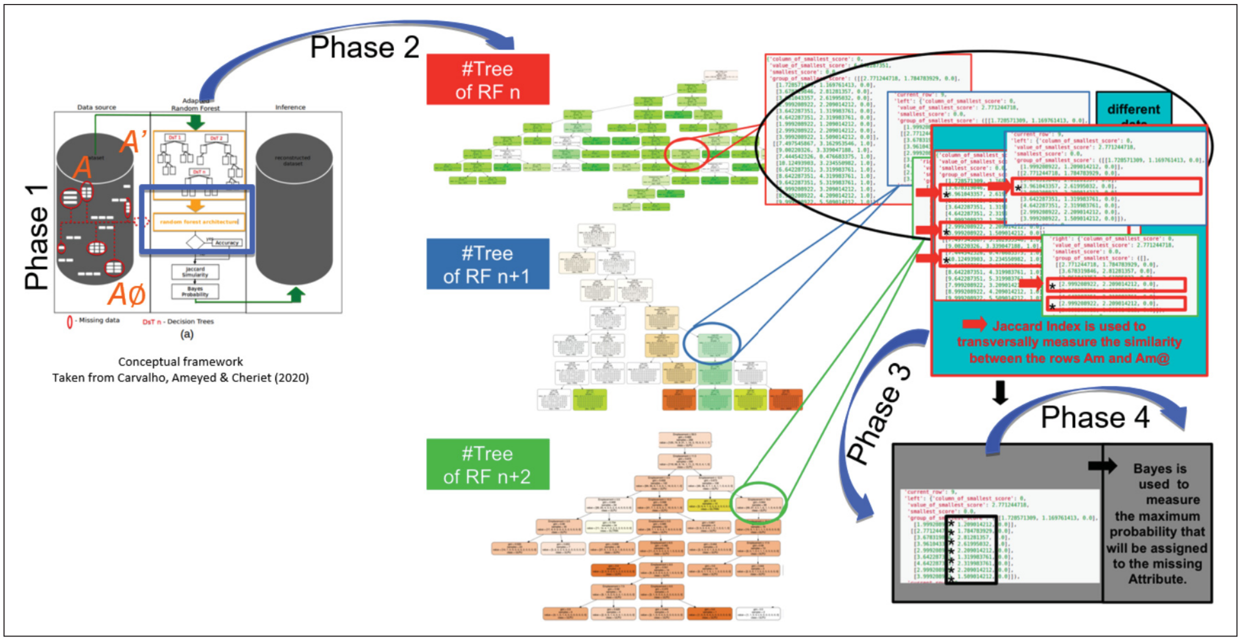


Figure 2.3 In phase 4, by applying Bayes probability, we will have the attribute with the maximum probability value to impute the missing one among all the most similar rows

2.2.1 Proposed data fusion framework

The sections below present in details the steps (preprocessing, data fusion semantic alignment, unsupervised deep learning, restricted boltzmann machines, NLP, and block-based structure matrix factorization) that comprise our proposed data fusion framework.

2.2.1.1 Ensemble deep learning model for data fusion

The task to deal with in our specific problem two, is to perform data fusion considering the dimensionality issue found in the real-world datasets. By proposing the fusion using the intermediate integration level, we can understand the inter/intra dataset relationships. For this aim, we introduce an innovative ensemble framework that matches a Restricted Boltzmann Machine (RBM) algorithm to perform feature learning and ensure that only significant ones will remain in the model, using a Natural Language Process (NLP) for the dataset that contains the text data type and a block-based matrix tri-factorization (MF) algorithm to deal with the fusion process.

2.2.1.2 Preprocessing

This stage represents the treatment of unnecessary attributes, distortions, redundancies, high-correlation data, and so on. Outliers are demonstrations of distortions, which might harm and change the numeric result measures as mean, median, variance, among others. Another kind of data that is useless in the analysis is ids attribute data, that contains a categorical meaning sometimes represented by an integer. Lastly, redundant data may affect the number of columns or rows in a dataset. In the first situation, it might yield the curse of dimensionality. Meanwhile, the second case might cause overfitting. According to, Alice Zheng (2018), these dataset obstacles can prevent a fair comparison between the observations. Both can severely jeopardize the learning process and decrease the next stage's deep learning performance. Fig.2.4 demonstrates a troublesome dataset. Thus, the framework receives each dataset D at the time, runs the preprocessing step, and deliver a cleaned-up version of the dataset W to the next phase.

2.2.1.3 Data fusion semantic alignment

Semantic alignment is required when the sources of data do not refer to a common object or phenomena. Otherwise if the inputs come from same sensors' type, observing the same object or phenomena, the semantic alignment is not necessary, Kuhn (2006). The semantic alignment is typically required in fusing processes of many different sensors and datasets and, in several cases, for the same phenomena, different sensors use either a different set of names or different symbols. It is known as "solving the label correspondence problem", Jegelka, Kapoor & Horvitz (2013). In this work we analyze our framework's performance by fusing geospatial dataset that relates various services provided in Montreal.

These data may or may not cover the town uniformly, therefore, understanding how these services could be related is an open question, although many research is undergoing to

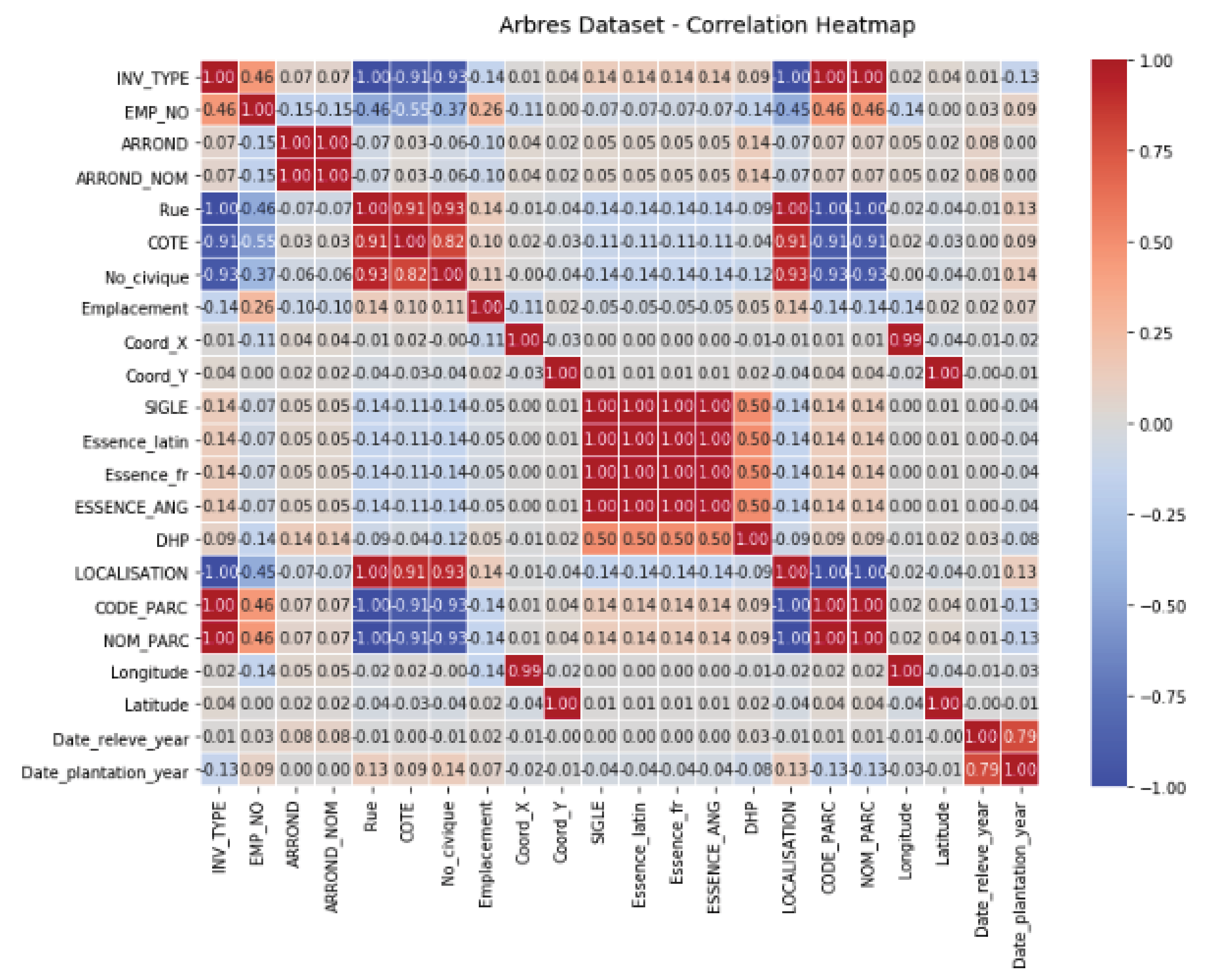


Figure 2.4   Correlation analysis in the preprocessing phase

have a cross-domain study, and subject of the interest of the decision-makers and smart city practitioners. We applied semantic alignment in the pairs of latitude and longitude, spatial labels of the dataset. For the matrix $A = (x1, x2,...,xn)$ the pair of labels attributes are extracted, $\alpha = \{(lat1,lon1),(lat2,lon2),...,(latn,lonn)\}$. For the correlated matrix $B = (x1, x2,...,xn)$, we extract the pair of labels attributes $\beta = \{(lat1,lon1),(lat2,lon2),...,(latn,lonn)\}$. Then, A and B are semantically aligned by finding the pair of labels α with their associated range of pair in β.

Assigning the correspondences in a new associated matrix *T*, we have:

$$T(\alpha,\beta) = \begin{cases} 1, & \text{if } \mathit{(lat,lon)}_{\alpha} \cap \mathit{(lat,lon)}_{\beta} \\ 0, & \text{otherwise} \end{cases}$$

Where *latn*, and *lonn* stand for latitude and longitude. In many applications, solving the assignment problem is the fusion algorithm itself, Mitchell (2010).

2.2.1.4 Unsupervised deep learning

Many DL algorithms are devised to learn about deep hierarchies of features from unlabeled data. In many cases, these algorithms involve multi-layered networks of complex features to train and require significant effort to tune. Consequently, applying DL methods is not a straightforward process because it demands a workaround to match the DL technique to obtain better results, Aldwairi *et al.* (2018). Due to these DL restrictions we clean, transform, and tidy the data in the preprocessing stage. Moreover, when dealing with datasets, it is usually necessary to find which data subset provides the learning model with better accuracy. In this sense, capturing the variability that describes most of the data means discovering its latent features. The latent features of every single dataset are prepared in this DL stage to the next fusion stage.

2.2.1.5 Restricted Boltzmann Machines

This process aims to find a compact latent representation of the numeric data in a nonlinear distribution. We use an RBM to discover the features of the data, which helps modeling the complex underlying relationships and patterns present in it. We learn the latent features, considering its hierarchy one level at a time. An incremental learning process adds one layer of weights to the RBM, which is an unsupervised, generative model, Mohamed & Hinton (2009). Layers in the RBM can be stacked and set with different numbers of nodes as shown in Fig. 2.5

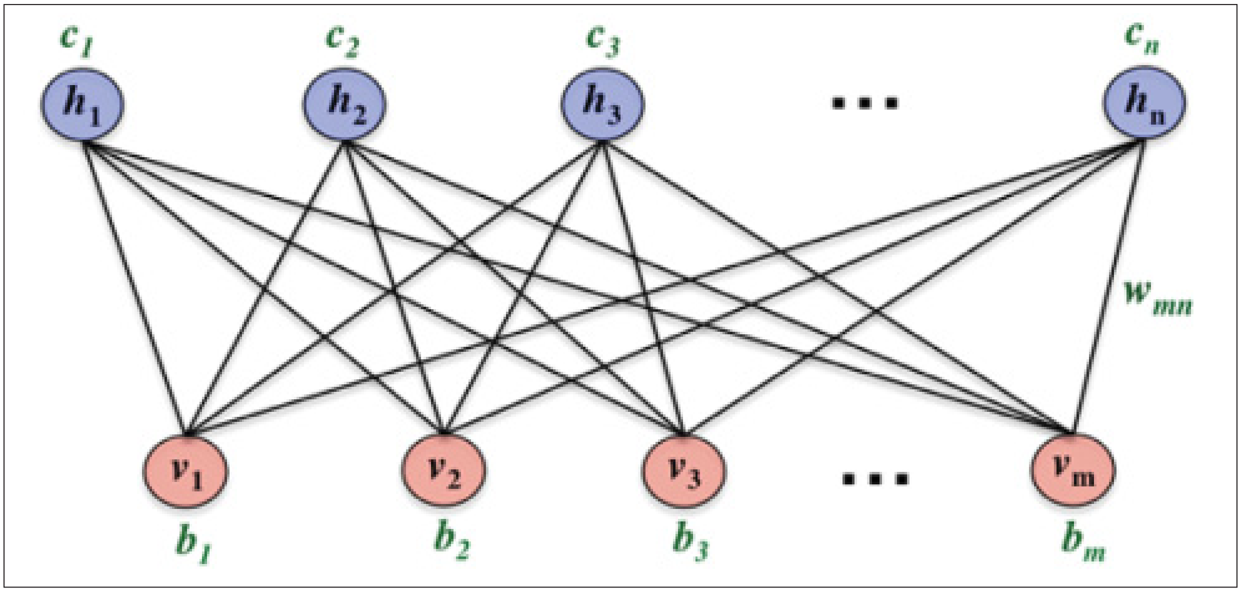


Figure 2.5 A graphical representation of an RBM with m visible and n hidden nodes Taken from Aldwairi *et al.* (2018)

When the quantity of nodes in the layer is reduced, the RBM operates similarly to the PCA algorithm. However, RBM has the capability to perform reduction as well as increase the number of nodes, such a flexibility of the RBM allow us to respond to the **research question 3**. The forward pass activates the hidden layer as follows:

$$p(h = 1|v) = \sigma(W^T v + c) \tag{2.5}$$

Where p are the probabilities of the hidden layer h given the visible layer v, respectively rows and columns of W, and W^T are the weights of each unit, while c is the bias term. In the backward pass, known as reverse phase, the inputs are reconstructed as:

$$p(v = 1|h) = \sigma(W^T h + c) \tag{2.6}$$

The conditional probabilities $p(h|v)$ and $p(v|h)$ give us the joint distribution $p(v, h)$. RBM's optimization algorithm tries to minimize the distribution differences between original input and reconstructed one, Hinton (2012). To do so, it uses KL-divergence to measure the dissimilarity between them. During the process of minimizing the error, rather than using all rows of the dataset, it requires just few samples. Thereby, performing Gibbs sampling from the distributions $p(v)$, and $p(h)$, it is obtained a sequence of rows that are approximated from a specified probability. It starts sampling $v(0)$ and based on this sample, the next sample $v(k)$ after k steps are produced. In $p(h)$, after t steps it is sampled $p(h|v(t))$ and $v(t+1)$ and subsequently $p(v|h(t))$. The algorithm to measure the divergence distribution is called Contrastive Divergence formulated as:

$$\begin{aligned} CD_k(W, v^0) = -\sum_h p(h|v^0)\frac{\partial E(v^0, h)}{\partial W} + \\ \sum_h p(h|v^k)\frac{\partial E(v^k, h)}{\partial W} \end{aligned} \tag{2.7}$$

This way we exploit the RBM as a feature learning algorithm able to receive clean datasets from the preprocessing stage and create brand-new features by taking advantage of latent structures within the data, Bengio *et al.* (2012).

2.2.1.6 NLP

Data obtained from text comments such as social media posts are unstructured data and it does not fit into the tabular row and column format. Texts represent massive volume of available data in the real world, and understanding the meaning, and latent semantics behind them enable us to improve the context-awareness in the fusion process, Mnih & Kavukcuoglu (2013). We use the matrix factorization model, Pennington, Socher & Manning (2014) to address adequately it in the unsupervised NLP context:

$$\hat{R} = W'U^T \tag{2.8}$$

Where *W'* is either the text data attribute or a set of text attributes inside of the datasets.*W'* contains the compact latent features matrix in the probability distribution *p(h* $=$ 1|*v).* $\hat{R}$

corresponds to the approximated model factorized in the NLP process, and we bring it for the next stage.

2.2.1.7 Block-based structure matrix factorization

Matrix factorization relies on a group of learning algorithms which can tackle latent variable models (LVMs). It deals with heterogeneous types of data, Murphy (2012) and works as a pattern recognition tool. Fig. 2.6 shows the conceptual fusion in a block-based structure. The algorithm is available in appendix III.

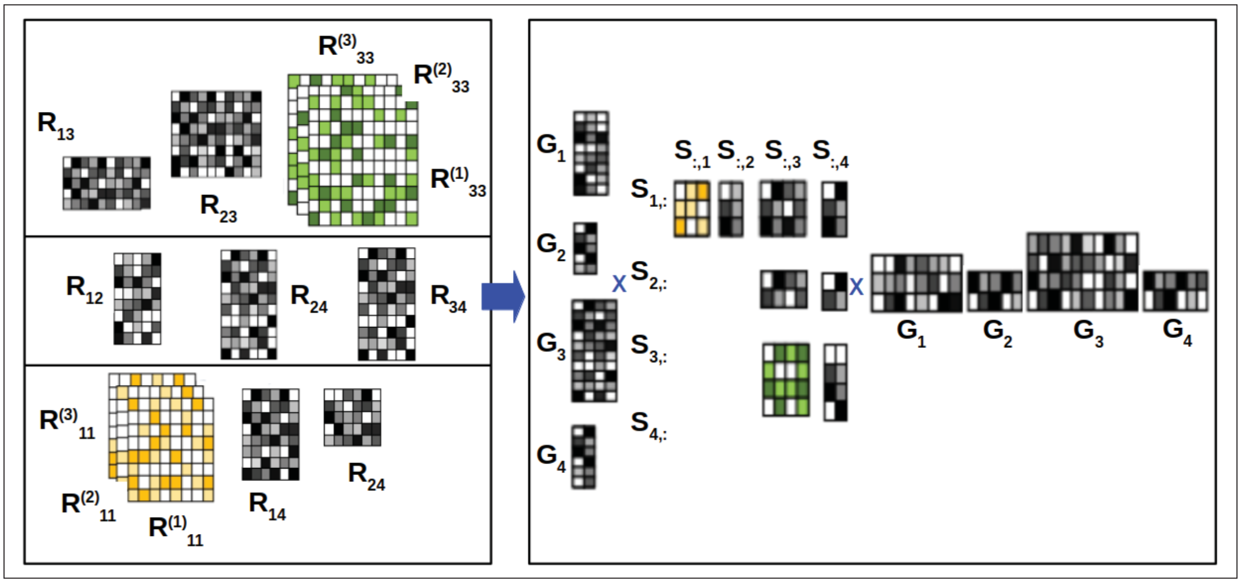


Figure 2.6 Conceptual fusion of block-based matrix structure
Taken from Wang *et al.* (2019)

In the left, R_{ij} represents the datasets. In the right side G_i and S_{ij} is the low-rank representation. When applying this technique, we accomplish the last two significant steps in our proposed framework: Load the compacted latent space matrices *R(s)* (that have been generated in the previous process and placed into the block-based matrix structure). Discover the patterns and correlations behind the whole process, according to , Zitnik & Zupan (2015) that has developed a tri-factorization penalized method which entirely solves the approximation of (2.9), as follows:

$$R_{ij} \approx G_i S_{ij} G_j^T \tag{2.9}$$

Where R_{ij} in R is the result of the inner product of the rows in matrix G_i, linearly combined with the columns in matrix S_{ij}, and is weighted by the correspondent column of G_j.

$$R = \begin{bmatrix} * & R_{12} & ... & R_{1r} \\ R_{21} & * & ... & R_{2r} \\ \vdots & \vdots & \vdots & \vdots \\ R_{r1} & R_{r2} & ... & * \end{bmatrix} \tag{2.10}$$

The outcome of such arrangement produces two kinds of factors. The first creates individual factors of each dataset, and the second produces shared factors used in the data fusion and detect patterns in it. The individual's factors are embedded into the matrix G while the shared factors are in S:

$$G = Diag(G_1^{n_1 x k_1}, G_2^{n_2 x k_2}, ..., G_r^{n_r x k_r}),$$

$$S = \begin{bmatrix} * & S_{12}^{k_1 x k_2} & ... & S_{1r}^{k_1 x k_r} \\ S_{21}^{k_2 x k_1} & * & ... & S_{2r}^{k_2 x k_r} \\ \vdots & \vdots & \vdots & \vdots \\ S_{r1}^{k_r x k_1} & S_{r2}^{k_r x k_2} & ... & * \end{bmatrix} \tag{2.11}$$

Datasets can contain similar data and relations, and the DFMF framework represents such datasets as $\theta_i^{(t)}$. These similarities are encoded in a set of constraint block diagonal matrices.

$$\theta^{(t)} = Diag(\theta_1^{(t)}, \theta_2^{(t)}, ..., \theta_r^{(t)}) \tag{2.12}$$

If the data are similar, the constraint $\theta^{(t)}$ is negative, being penalized in the cost function in the calculation of G and S, and positive if there are no similarities in the datasets. Thereby, the DFMF minimizes the objective function as can be seen next:

$$\begin{aligned} \underset{G\geq 0}{min} J(G;S) = \sum_{R_{ij}\in R} \parallel R_{ij} - G_i S_{ij} G_j^T \parallel^2 \\ + \sum_{t=1}^{max_i t_i} tr(G^T \theta^{(t)} G) \end{aligned} \tag{2.13}$$

The process flow and the concepts discussed in this section are show in Fig. 2.7

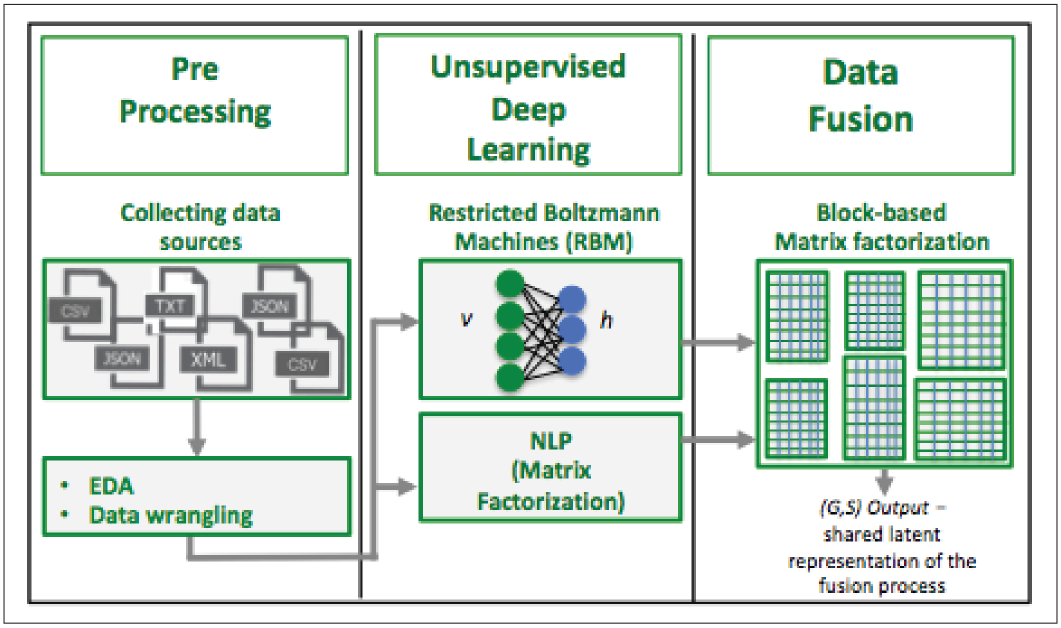

Pre Processing
Unsupervised Deep Learning
Data Fusion
Collecting data sources
CSV
TXT
JSON
JSON
XML
CSV
• EDA
• Data wrangling
Restricted Boltzmann Machines (RBM)
v
h
NLP (Matrix Factorization)
Block-based Matrix factorization
(G,S) Output – shared latent representation of the fusion process

Figure 2.7   Conceptual framework's structure

2.2.2 Proposed data fusion algorithm

We show in detail the proposed framework presented in section 2.2 and discuss the typical complexity of fusing multiple heterogeneous datasets. The notations for all inputs and parameters used in the thesis are summarized in Table 2.2. Regarding categorical values, in real-world datasets, usually large numbers of categorical columns are found. When these contain high

Table 2.2 Inputs and parameters notation

Inputs Notations	**Description**
D	Dataset in a m x n matrix format
W	Cleaned dataset after the preprocessing
W’	Set of attributes with text values inside of the datasets
$\hat{W}$	Reconstructed datasets in the RBM process
R	Compact latent features matrix (inputs for the fusion process)
$\hat{R}$	Approximated model factorized in the NLP process
G_i	Used to reconstruct each dataset
S_{ij}	Latent components interaction in the respective relation

cardinality (many different values), the traditional hot encoders found in the features engineering literature (e.g., one-hot encoding, count, or frequency encoding) lead to many new columns. It likely affects the model due to the curse of dimensionality or loses a considerable amount of original information, Schafer & Graham (2002).

Frequently, it is necessary in the presence of numeric values to employ some feature transformation strategy, such as normalization, scaling, or binning, according to the numeric type (continuous or ordinal). This may affect the ML results. Thus, it demands special attention in the preprocessing phase of the ML pipeline. The main steps of algorithm 2.2 are:

- the set of the related heterogeneous datasets D, are inputs of the framework. These can be considered asymmetric datasets (different dimensions).
- each dataset D_i is preprocessed to be clean and tidy. The datasets analyzed in this stage are highlighted from lines 6 to 22.
- we store each dataset, renaming then to W. If we find text attributes in the dataset, they are stored as W’. (lines 26 and 30).
- in the unsupervised deep learning phase of the framework we use two different techniques based on each type of data. For non-text content (in the cleaned datasets W) we apply an RBM algorithm to retain the latent representation of them. The output of this step is the compact learned version of each dataset, named as R. (line 41).
- for text content (also in the cleaned datasets W’) we apply the GloVe algorithm. The output of this step is the factorized matrix factors, assigned as $\hat{R}$. (line 55).

Algorithm 2.2 Framework to fuse multiples heterogeneous datasets

```
Input: set of $D$
Result: $G;S$
for $i$ in len(D-1) do
    $preproc\_finish = false$;
    $D_i \leftarrow D[i]$;
    $k = [2, 5, 10, 20, 50, parK]$;
    while preprocessing steps do
        $D_i \leftarrow$ identify attributes type subprocess();
        $D_i \leftarrow$ treat ids attributes subprocess();
        $D_i \leftarrow$ treat duplicates attributes subprocess();
        $D_i \leftarrow$ missing values subprocess();
        $D_i \leftarrow$ embed categorical types subprocess();
        $D_i \leftarrow$ treat correlated attributes subprocess();
        if $preproc\_finish == true$ then
            compute $W_i \leftarrow D_i$;
            if $text\_attrb$ in ( $D_i$ ) then
                compute $W_i' \leftarrow D_i$;
            end if
        end if
    end while
    $rmseDs = []$;
    $rmseWs = []$;
    for $j, v$ in enumerate(k) do
        $R_j \leftarrow$ call RBM fit ( $D_i, v$ );
        $rmseDs[j]$ = call Rmse ( $D_i, R_j$ );
        for $c$ in len(rmseDs) do
            if $rmseDs[j] < rmseDs[c]$ then
                compute $R[i] = R_j$;
                compute $kFinal[i] = v$;
            end if
        end for
        $\hat{R}_j \leftarrow$ call NLP fit ( $W_i', v$ );
        $rmseWs[j]$ = call Rmse ( $W_i', \hat{R}_j$ );
        for $c$ in len(rmseWs) do
            if $rmseWs[j] < rmseWs[c]$ then
                compute $\hat{R}[i] = \hat{R}_j$;
                compute $kFinal[i] = v$;
            end if
        end for
    end for
    $(G;S) \leftarrow$ call DFMF $(R[i], \hat{R}[i], kFinal)$;
end for
```

- next all R and $\hat{R}$ are loaded in the block-based matrix structure as the input for the DFMF algorithm, which simultaneously factorizes all datasets in the fusion process. The output matrices G and S are employed as a shared latent representation of the fused inputs. Line (70).

2.2.2.1 Discussion

We bring a complete framework that can deal with feature in-feature out (FeI-FeO), Dasarathy classification in intermediate integration level through the semantic alignment principle. To make it possible, following the procedure with the proper treatment in datasets, we applied an RBM model to transform raw attributes in features that capture most of the latent representations of information. The GloVe tool is used to tackle textual data due to the massive amount of this type of material in our use case. Consequently, we fed the DFMF framework to fuse the input features and receive the shared matrix factors which represent the whole structure.

This proposed framework can positively be the answer to the **research question 4**. Throughout of the process steps described above, with the proposed ensemble deep learning model for data fusion we can fuse heterogeneous datasets. Our inputs are individual datasets from different domains and our final output are the matrices G and S which share the latent representation of the fused inputs. In Fig. 2.8, we synthesized the mechanism behind the algorithm.

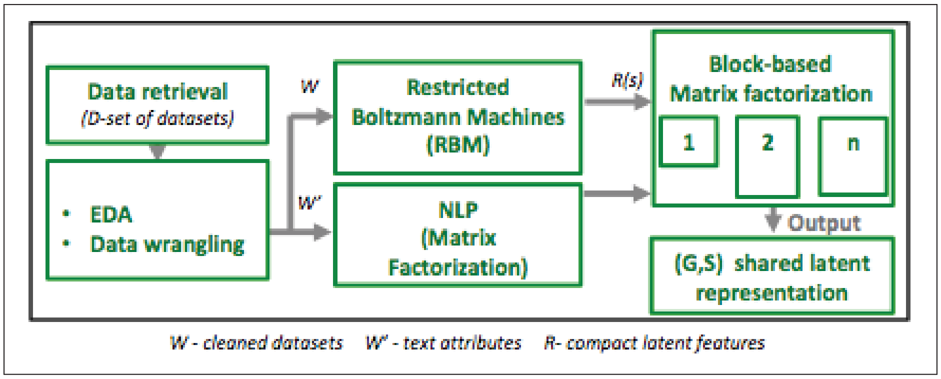

Data retrieval
(D-set of datasets)
EDA
Data wrangling
W
W'
Restricted Boltzmann Machines (RBM)
NLP (Matrix Factorization)
R(s)
Block-based Matrix factorization
1
2
n
Output
(G,S) shared latent representation
W - cleaned datasets
W' - text attributes
R- compact latent features

Figure 2.8 Synthetized framework view

2.3 Conclusion

As previously stated in the thesis objective, our primary goal is to perform KDD by fusing multiple datasets. However, tackling the missing data problem found in the real-world dataset is a mandatory step to its success and so, we broke down the objective into two sub-objectives. In the first part, to address the widespread missing values problem, we have proposed an ensemble learning for heterogeneous missing data imputation capable of addressing various data types, both in labeled and unlabeled datasets. The framework is composed by combining the random forest, one of the most widespread and robust ML algorithms, with the Jaccard index and Bayes probability to impute missing values based on data topology, Albergante, Mirkes, Chen, Martin, Faure, Barillot, Pinello, Gorban & Zinovyev (2018), Mohan & Pearl (2018).

In the second part, to overcome a current data fusion problem linked to the heterogeneity in the real-world dataset, we presented a complete framework that can deal with feature in-feature out, Dasarathy classification in the level of intermediate integration. To make it possible, after proceeding with the proper treatment in datasets, an RBM model has been employed to transform raw attributes in features that capture most of the latent representations of information. Meanwhile, the GloVe tool is used to deal with textual data due to the massive amount of this type of material in our use case. Lastly, we fed the DFMF framework to fuse the input features and receive the shared matrix factors, representing the whole structure. We have tested both proposed frameworks presented in this methodology chapter, and the results are shown next.

CHAPTER 3

EXPERIMENTS AND VALIDATION RESULTS

In this chapter we demonstrate the experiments setup and the results of the ensemble learning for heterogeneous missing data imputation and ensemble deep learning model for data fusion proposed in sections 2.1.1 and 2.2.1, part I and part II, respectively. In the first part, we state the experimental protocol, the datasets in the case study, and our evaluation criteria. Then, we show and discuss the experiments results from part one. In the second part, we introduce the datasets in the case study, the evaluation criteria for the training, testing, and fusion phases of the framework, then finally we present and discuss the results.

3.1 Part I - Missing values imputation performance evaluation

3.1.1 Experimental protocol - baseline

We compare our framework's performance versus statistical techniques and model-based methods of imputing missing values.

- *Comparison with statistical techniques:* The set of techniques extensively used as a baseline in the statistical approach to continuous and discrete data types are mean and median imputation. MFV is a baseline to categorical and binary, Musil, Warner, Yobas & Jones (2002). The mean, median, and MFV replace the missing values with the local or global mean, median or MFV of the attribute of the set with missing values. These baselines are fast and easy to implement and can quickly scale with the dataset size. The drawback of these univariate imputation techniques is that it introduces bias and harms the statistical strength.
- *Comparison with model-based approaches:* The GLRM framework, which deals with heterogeneous data types, is implemented in relevant standard ML and database platforms as Spark, Python, Julia, and R. PCA is a well-known technique that minimizes the best rank-k using the least-square. Due to the possibility of dimensionality reduction, it is possible to be

used to input missing values. Although it is a statistical technique, its properties are widely used in many ML algorithms to deal with continuous values.

3.1.2 Datasets in the case study

In this work we are mainly exploring the variety and complexity of big data while experimenting with the IoT dataset, the images dataset, and datasets with heterogeneous data types. The volume is addressed in the experiment with a dataset that contains up to 310.000 rows; velocity is not the focus of the present thesis. We evaluated our proposed framework extensively on five different datasets that contained labeled and unlabeled data, as it follows:

1. the air quality (IoT) dataset of the city of Montreal available in the Smart City's open data platform. It contains the air quality of city, regarding five pollutants: Sulfur dioxide (SO2), Carbon monoxide (CO), Ozone (O3), Nitrogen dioxide (NO2), and breathable fine particles (PM). It has 428 rows and five attributes (continuous, discrete, and categorical) in which, 128 rows (30%) there are missing values in at least one attribute, it is a MCAR case.
2. public trees of the city is the dataset that catalog the trees in parks and streets of Montreal. From 316.070 rows, 220.027 (69.61%) represents the trees in the park and 96.043 (30.39%) represents the trees in the streets, a MNAR case. It has 22 categorical and discrete attributes.
3. the synthetic dataset contains 50 rows and 60 attributes (continuous, categorical, and binary) generated by the standard normal distribution.
4. the fashion MNIST is a well-known image dataset in ML community, with 60.000 rows in the training and 10.000 rows in the test set with 784 continuous attributes. We decided to evaluate the framework in the scenario of a unique data type.
5. the Titanic open dataset is a well-known in the ML community, with 891 rows and 12 attributes (discrete and categorical), it is a MAR case. The specifications of these datasets are summarized in Table 3.1.

In this experiment protocol, considering that dealing with the critical missing data problem found in the real-world datasets is a key factor to an effective and efficient fusion, we have chosen five datasets to ensure the capacity and extensibility of our framework to process heterogeneous datasets with different data types. More specifically, our focus in this phase is to provide a broad framework able to process heterogeneous raw data. There is no inner relation among the datasets and no connection among them with the smart environment. However, the datasets air quality and public trees represent services of the city, and they are hosted in the Montreal open data platform.

Table 3.1 Dataset details

Dataset	Data type	Dimensions	Type
Air quality	Continuous/Categorical.	428 x 5	Unlabeled
Public Trees	Continuous./Categorical.	316.070 x 22	Unlabeled
Synthetic Data	Continuous./Categorical./Binary.	50 x 50	Unlabeled
Fashion Mnist	Continuous.	60.000 x 748	Labeled
Titanic	Continuous./Categorical.	780 x 22	Labeled

3.1.3 Evaluation criteria

For continuous and discrete values, we used the root mean square error (RMSE) (3.1), and coefficient of determination R2 (3.4) to evaluate the framework's performance compared to other techniques for the continuous and discrete values. RMSE is a popular standard used to measure the error of the model for this type of attribute. There are continuous values with different scales in the datasets and due to it we use R2, defined by the sum of the squared errors (SSE) divided by the total sum of the squares (SST). It usually normalizes the scales, but not necessarily from 0 to 1 and can assume negative values for example in non-linear functions.

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(a_m - a_m\emptyset)^2}{n}} \tag{3.1}$$

$$\mathrm{SSE} = \sum_{i=1}^{n}(a_m - a_m\emptyset)^2 \tag{3.2}$$

$$\mathrm{SST} = \sum_{i=1}^{n}(a_m - mean(a_m\emptyset))^2 \tag{3.3}$$

$$\text{R2} = 1 - \frac{\text{SSE}}{\text{SST}} \tag{3.4}$$

For the categorical, and binary attributes we used the misclassification error (3.5) as the metric to evaluate the performance of our framework.

$$\text{Miss. Error} = \sum_{i=1}^{n} \frac{1, (a_m = a_m \emptyset); 0}{n} \tag{3.5}$$

$$\text{D}_{\text{KL}}(\text{P}||\text{Q}) = \sum_{i} P(i) Log \frac{P(a_m)}{Q(a_m \emptyset)} \tag{3.6}$$

$$\text{D}_{\text{JS}}(\text{P}||\text{Q}) = \frac{1}{2} D(P(a_m)||M) + \frac{1}{2} D(Q(a_m \emptyset)||M) \tag{3.7}$$

$$\text{M} = \frac{P(a_m)}{Q(a_m \emptyset)} \tag{3.8}$$

Where DKL and DJS stands for Kullback-Leibler and Jensen-Shannon dirvergence, respectivelly. The operator || indicates divergence; P divergence from Q.

3.1.4 Experiments and results

In the learning phase, we performed the experiments before the imputation using the parameter *ndt* (see Table 2.1) with 5, 10, and 15 trees for each dataset. In all of them, we ran five folds cross-validation partition, with 70% of subset *A'* as training to fit the model and 20% as validation data to fine-tune the model. In the test phase, we ran the experiments 11 times, randomly removing the ground truth values of the attributes varying from 1% to 34%. Fig. 3.1 shows the RMSE results for the 11 experiments in which attributes with missing values were increased from 1% to 34%. Examining the RMSE, as expected, it has grown for all techniques. However, the proposed framework seems to outperform other players in all experiments. The RMSE

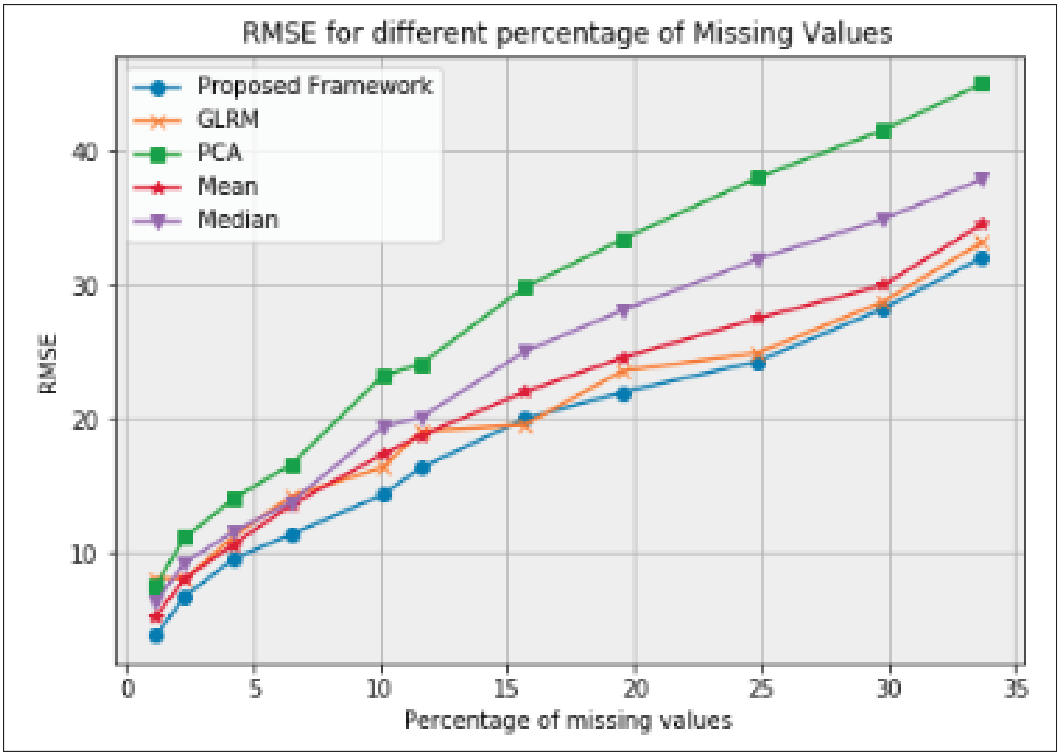


Figure 3.1 RMSE for the experiments with continuous and discrete values in the datasets

of our framework varied from 3.81 to 32.01 and the GLRM RMSE range varied from 8.13 to 33.17. It closely follows the proposed framework from the stage in which the percentage of attributes with missing values reaches 25%. The imputation was also performed for the following statistical techniques: mean, median, and PCA respectively. Fig. 3.2 shows the R2 metric for each technique. This second metric reinforces the advantage of our framework over the others techniques, and as can be seen our framework reached a peak of 1.0 and a minimum of -1.04, with a mean of 0.47, the best imputation efficiency. Second is the GLRM, which ranges from 0.94 to -1.00 with a mean 0.36. The R2 metric for the mean, median and PCA are 0.40, 0.23, and 0.08, respectively. This figure highlights outliers present in all techniques. Our proposed framework has fewer outliers and imputs missing values more uniformly thanks to the Bayes probability that performs with multiple data distribution. Considering all experiments, the average performance of the framework is 11.54% above the GLRM, 18.62% better than the mean, and 35.89% and 54.86% higher than median and PCA, respectively.

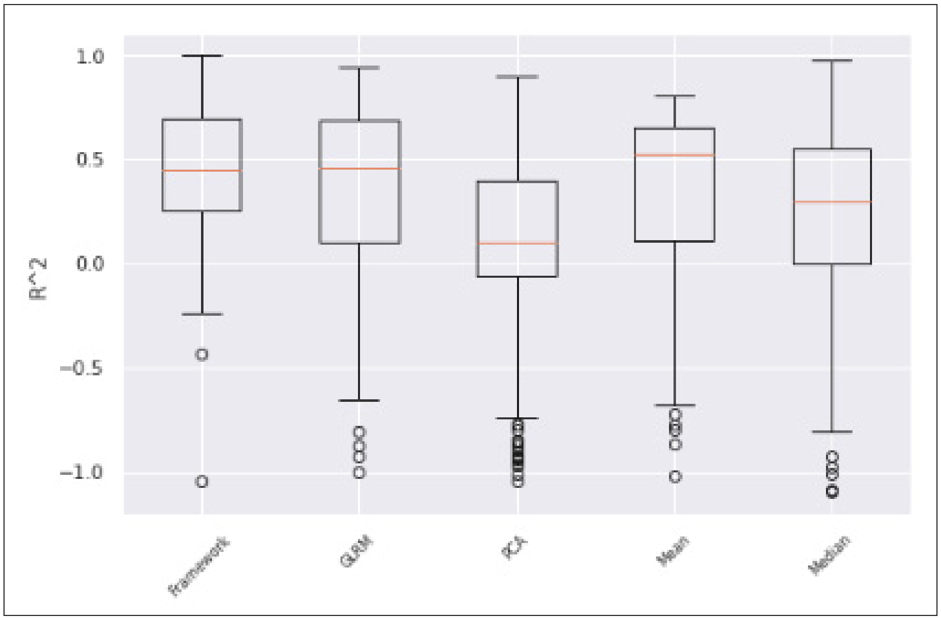


Figure 3.2 R2 results for the experiments with continuous discrete values in the datasets

Fig. 3.3 offers the misclassification error for categorical and binary values. Clearly, the proposed framework achieves better results than the GLRM and MFV. The minimum result reached was 0.67 and the maximum was 0.85 for 2% and 17% of the missing values respectively. Considering the combination of our evaluation criteria (1-correct, 0-incorrect) and the low variance, all three methods struggled in the case of a small amount of missing values. Although, even in this case our framework surpassed the other techniques. At the stage of 20% of missing values it is possible to see that it is fairly stable.

Fig. 3.4, shows the distributions of the categorical attributes in different datasets. As can be noticed, the attributes are non-normal distributed. In (a) the Kullback-Leibler (3.6) and Jensen-Shannon divergences (3.7) between the original distribution, our framework, the GLRM, and MFV are, respectively, 28.64, 37.85, and 69.27 for a categorical attribute in Air quality dataset. In (b), similarly, the Kullback-Leibler divergences for the three techniques are 34.00, 47.16, and 56.49 for the Public trees. Finally, in (c), for a categorical attribute in the Titanic dataset, the Kullback-Leibler divergences are 480.45, 510.93, 686.78. The average performance of the framework in imputing categorical and binary type is 13.99% and 69.06% superior to the

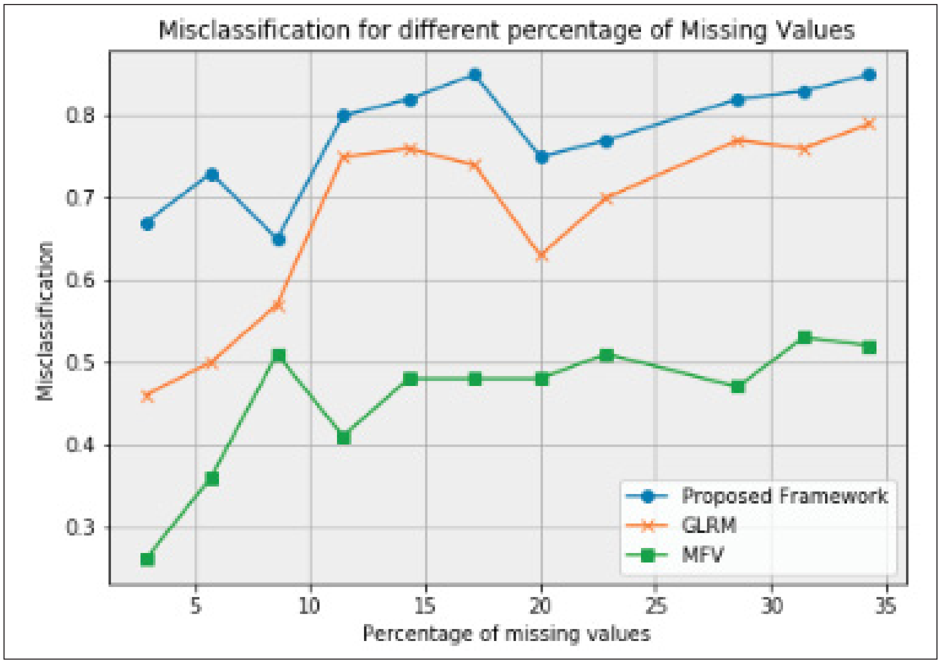


Figure 3.3 Misclassification error for the experiments with categorical and binary values

GLRM and MFV. In general, the goal for the real-world dataset that suffers from the missing values problem, is to retrieve the data in a manner that it retains, as much as possible, the statistical power of the dataset. In other words, it can keep the conclusions drawn from the data.

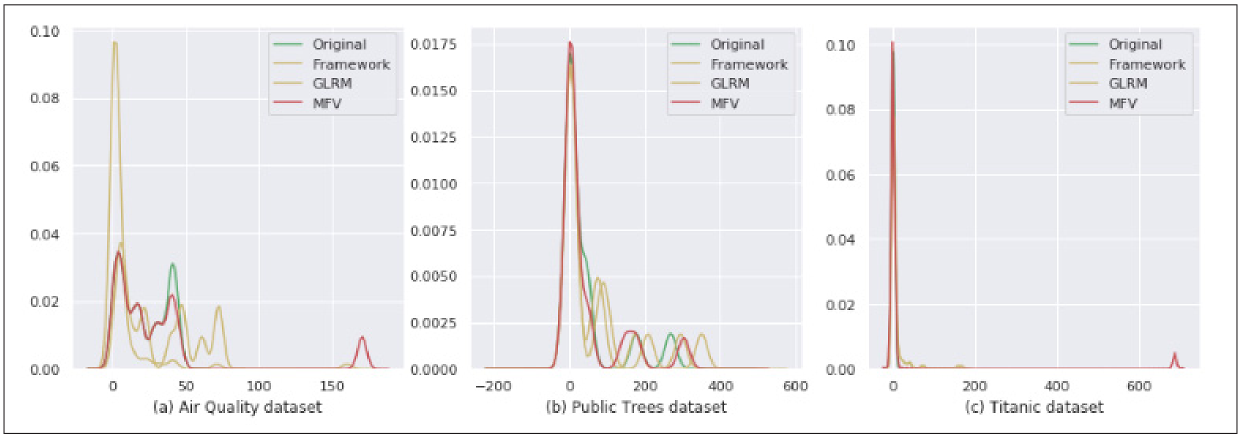


Figure 3.4 Feature distribution

In the left figure, we present the distribution of the air quality dataset with the original 337 rows without missing values. In the middle figure, we present the distribution of the public trees dataset with the original 220.027 rows without missing values. In the right figure we present the distribution of the Titanic dataset with the original 183 rows without missing values. In Table 3.2, it is possible to analyse that the proposed framework provides the smallest divergence distribution, standard deviation, and mean, as compared to the other methods.

Table 3.2 Further statistical comparison for the categorical imputation

-	Dataset	KL Divergence	Jensen Shannon Divergence	Standard Deviation	Mean
Framework	Air quality	28.64	0.093	24.88	30.5
	Public Trees	34.00	0.099	87.84	54.28
	Titanic	480.45	0.578	19.61	7.375
GLRM	Air quality	37.85	0.189	27.45	32.17
	Public Trees	47.16	0.105	83.12	52.12
	Titanic	510.93	0.773	35.04	13.45
MFV	Air quality	69.27	0.251	41.36	34.5
	Public Trees	56.49	0.162	84.2	47.5
	Titanic	686.78	0.902	146.08	33.71

Overall, the results shows that our framework has obtained a small RMSE for continuous and discrete values, and it was able to preserve the similar distribution of imputing categorical and binary attributes compared to the benchmark.

3.1.4.1 Discussion

We validated our framework using five datasets of different sizes and heterogeneous data types. The framework performed 11.54% superior to the GLRM, our benchmark in the model-based approach, and 18.62% better than the mean, the common statistical technique used. For categorical and binary types, our framework achieves performance that is 13.99% and 69.06% higher than the GLRM and MFV. Ultimately, it shows capacity to enhance the power of analysis and contributes to improve data quality by accurately reconstructing the dataset to impute missing values appropriately. Hence, it can be employed in a wide range of datasets for supervised and unsupervised tasks. It is worthwhile to highlight that our approach addresses the categorical type as it is, without further transformation or encoding treatment. Based on these results, we can assert that our proposed framework allow us to meet the requirements established in our specific objective **SO1**.

3.2 Part II - Data fusion

Integrating data and correlating them, utmost can represent the set of services offered in the city, considering that analyzing each domain as a silo or vertical of information configures a limitation, and data fusion can tackle it.

3.2.1 Data fusion framework performance evaluation

3.2.2 Datasets in the case study

To ensure the capability and scalability of our framework when performing in heterogeneous datasets, we highlight six domains: infrastructure, environment, natural resources, transport, health, law/Justice, and public security of Montreal, as summarized in Table 3.3. We choose eleven datasets as described in Table 3.4. These datasets help the understanding as well the correlation of mobility, health, and environment, among other domains in Montreal. Only if data from multiple domains are combined and correlated, we can have a holistic view of the city that holds together services, and the processes necessary to either to a single person or for an entire community.

3.2.3 Evaluation criteria

The RMSE (3.9) root mean square error has been adopted in order to evaluate the framework's performance. It is a popular standard metric to measure the model error in unsupervised learning tasks. According to, Meng *et al.* (2020) in a data fusion context, RMSE also analyzes the impact of the fusion algorithm and the degree in which the fusion model improves the information

Table 3.3 Dataset details

Dataset	**Domain**	**Dimensions**
Trees Inventory	Environment Natural Resources and Energy	316.070 x 22
Towing of vehicles	Infrastructure	134534 x 13
Water distribution	Environment Natural Resources and Energy	18029 x 13
Public Wi-Fi	Infrastructure	443 x 9
Taxi stands	Transport	402 x 10
Public parking lots	Infrastructure	75 x 15
Traffic surveillance cameras	Infrastructure	522 x 16
Health priority inspection	Health	69408 x 9
Traffic lights	Infrastructure	969664 x 30
Bedbug elimination	Health	37355 x 13
Criminal acts	Law-Justice and Public-Security	168139 x 8

reconstruction. A smaller RMSE means lower bias between ground truth data and fused results, leading to better fusion quality.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{M_1}(R_{m1} - \hat{R_{m1}})^2 + ... + \sum_{j=1}^{M_k}(R_{mk} - \hat{R_{mk}})^2}{M_1...M_k}} \tag{3.9}$$

3.2.4 Experiments and Results

3.2.4.1 Training phase

The two most influential aspects in this step are the quality of the data that supplies our RBM model and the hyperparameters' setup. The steps to ensure the quality of the data have been previously explained, and the RBM performance depends on a fine tune of hyperparameters that fits better with each dataset, considering that we are in a scenario where multiple and heterogeneous datasets are being processed. To optimize the RBM results, the hyperparameters batch size, iterations and learning rate should be tuned. For this, some preliminary tests have

Table 3.4 Datasets chosen in the fusion process

Dataset - ds	**Description**
ds1 - Trees Inventory	It catalogues the trees in parks, streets, and other public places.
ds2 - Towing of vehicles	It brings the tows carried out due to obstructions in the operations of the town, snow removal, roadwork, or even over special events.
ds3 - Water distribution	Used to carry out water balances and allow the measurement of the distributed water to other cities of Montreal Metropolitan Area.
ds4 - Public Wi-Fi	There are free wireless networks with both outdoor and indoor coverage in public buildings.
ds5 - Taxi stands	They ensure the access to fast and efficient service.
ds6 - Public parking lots	in these free parking lots, cars are reallocated during snow removal operations.
ds7 - Traffic surveillance cameras	The cameras are installed at intersections to provide real-time monitoring. They inform the public about traffic conditions and the possibility to either change their mode of transportation or their itinerary.
ds8 - Health priority inspection	To point out the addresses with high probability to develop health issues so they can be inspected by a team from the Housing and Development department.
ds9 - Traffic lights	Used to count vehicles, trucks, cyclists, and pedestrians at the most critical intersections which are equipped with traffic lights.
ds10 - Bedbug elimination	The bedbug exterminations are reported by the pest managers.
ds11 - Criminal acts	It contains the police service of the city of Montreal (SPVM) recorded acts.

been performed in order to limit the values search space because it is not feasible to test all possible values expecting to find those that will best satisfy the set of hyperparameters.

After restricting the potential values for one hyperparameter, we keep it constant and change the others. Starting from the learning rate, we have observed that the proper subspace values varies from 0.001 to 0.1. After discovering the learning rate that minimizes the reconstruction error for each dataset, we continued using it to estimate the optimal iteration and batch size. In Fig. 3.5, it is shown the RMSE for each dataset with its tuned learning rate, iteration and batch size. To estimate the error, twenty independent runs have been done with random seeds. 70% of the data was used to training, and the remaining 20% of it has been hidden for the validation phase. As expected, each dataset had different behaviours. The datasets 3, 6, 7, and 11, Water distribution, Public parking lots, Traffic surveillance cameras, and Criminal acts, respectively, have the smallest RMSE with learning rate value of 0.001. Meanwhile, the Towing of vehicles, Public Wi-Fi, and Bedbug elimination, datasets 2, 4, and 10 reached the smallest RMSE with a learning rate of 0.1. The other datasets had their smallest RMSE in diverse scales of the learning rate range. For instance, the dataset 1, Trees inventory, presents the smallest value with learning rate of 0.067. This analysis allows us identify the datasets differences and how each one deserves a specific treatment in order enhance the fusion process. From (3.9), datasets RMSE is 0.18, meaning that after tuned with the chosen hyperparameters the RBM, the model achieved excellent results. The results might suggest (to some extent) the quality of the model in the training stage, although this does not ensure that the model will perform the same way in the presence of the unseeing data, which is the aim of generalization power. Overall, the model

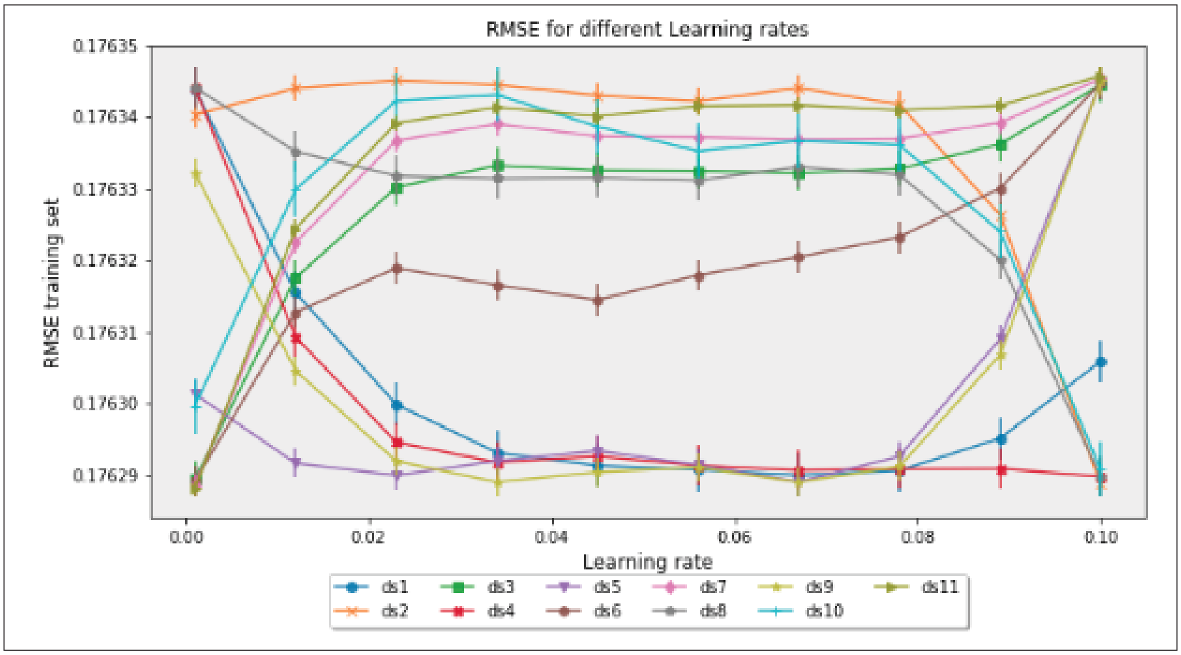


Figure 3.5 Learning rate varying from 0.001 to 0.1

performs as good as it is expected at this phase of trainning and validation.

3.2.4.2 Test phase

In this step, Fig. 3.6 illustrates the RMSE evaluation for the 10% remaining data, using the best hyperparameters found in the training phase. We obtained more comprehensive results, for example, in the Public parking lots dataset 6, the value was 0.186 for the test set and 0.176 for the training. This result meets the expected performance for this phase; here, the RBM works on an unseeing dataset's part, which was entirely omitted from the training stage. The RMSE for the set of the datasets is 0.206 that is 10% larger than in the training phase. All these results allow us to conclude that RBM illuminates the latent features representation used in our ensemble learning model, avoiding the bias and overfitting problems.

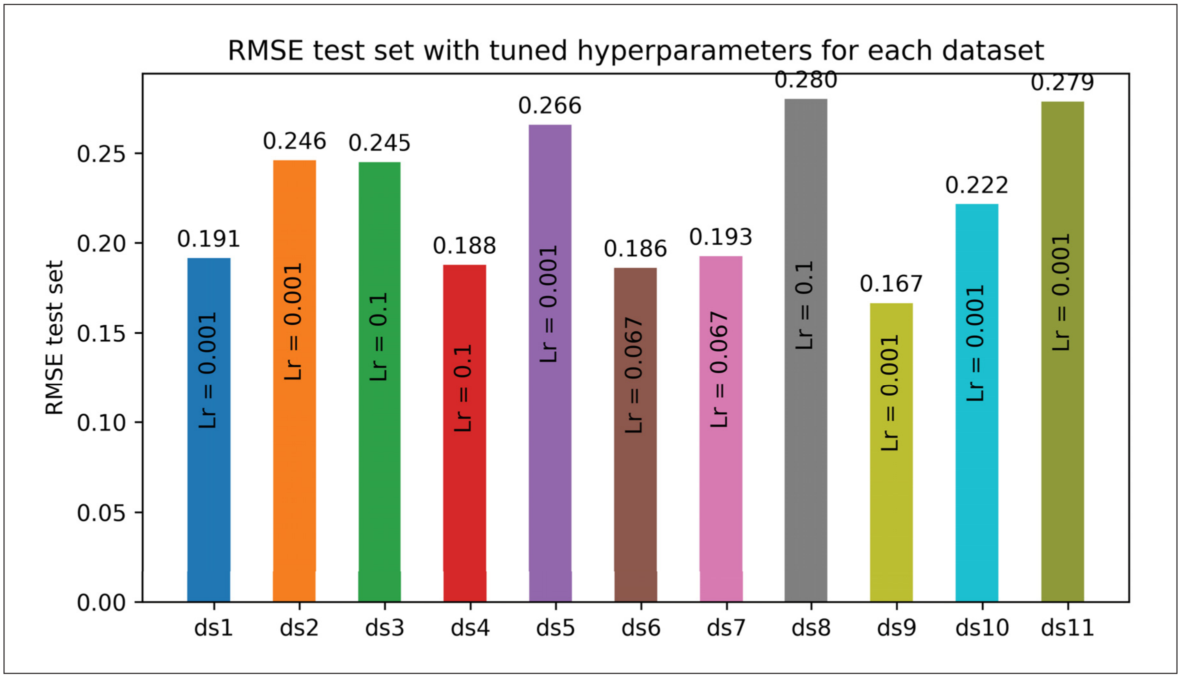


Figure 3.6 RMSE for the test set of each dataset

3.2.4.3 Fusion phase

The depurated dataset with latent features received from the previous process was aligned in the block-matrices structure. Ten runs have been conducted, varying the iteration of the DFMF algorithm from 10 to 200. Fig. 3.7 shows the RMSE subtly grew according to the dataset addition. When we started the fusion process with dataset one, the Trees inventory has obtained a RMSE of 0.135 while the whole fusion process with the 11 datasets was 0.163. Such results are consequence of block-matrix structure, from equations 5 to 9 which retains meaningful information exploited in the intermediate fusion process chosen here.

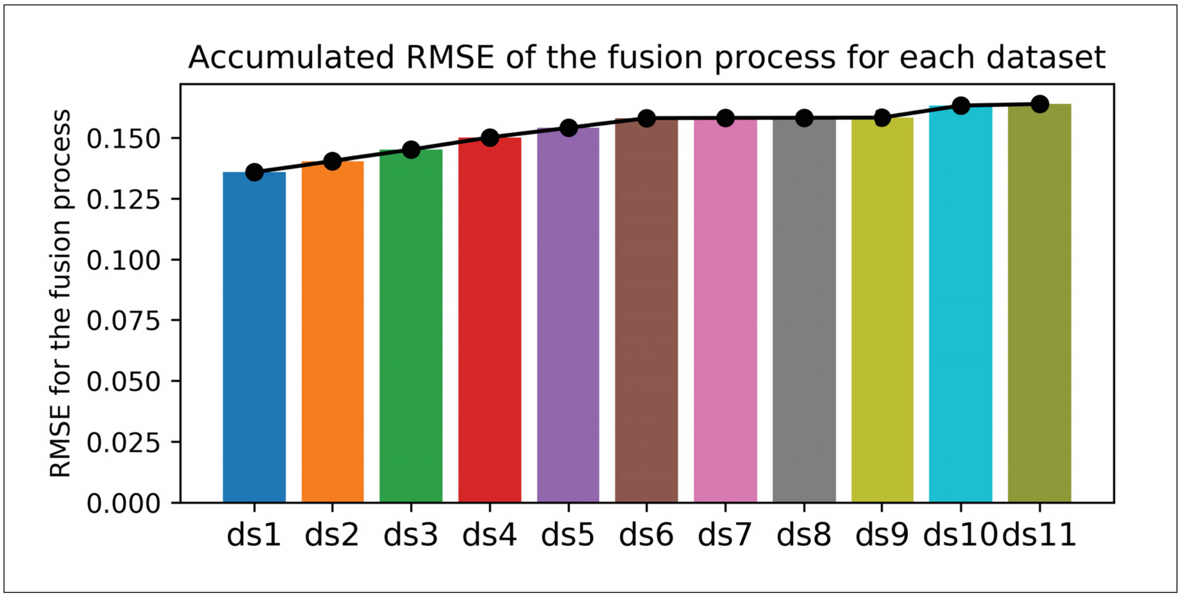


Figure 3.7 Accumulated RMSE for Fusion process according to dataset addition

The accumulated RMSE for each dataset added in the model is depicted in Fig. 3.8. Therefore, considering the RMSE metric, the fusion process was efficient to rebuild input matrices structure with reduced loss. This phase outcomes are the factor matrices G and S. Through these shared factor matrices, the interconnected patterns of the datasets are finally revealed.

Fusing real-world heterogeneous datasets is a challenging process, where the main goal is to understand the whole scenario with information from multiples sources. Fig. 3.9 shows clearly

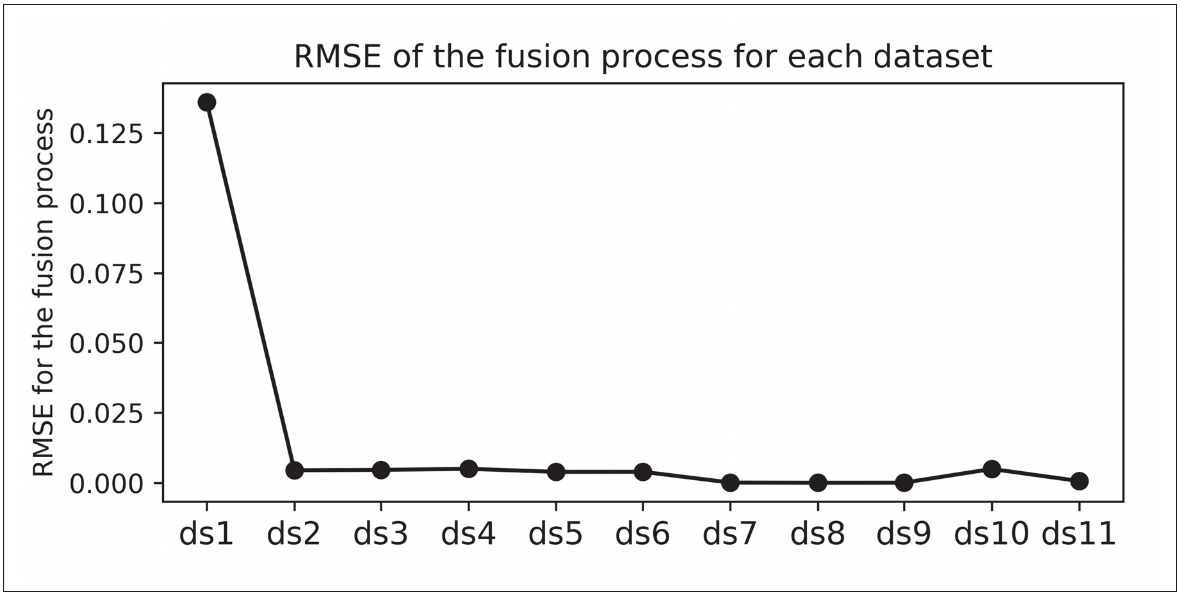


Figure 3.8 RMSE for Fusion process of each dataset

the inter/intra connection of the datasets adopted in our use case. This graph allows us to make distinct analyses such as the positive correlation of 0.58 in the relationship between the dataset one and six, Trees inventory and Public parking lots, that corroborates the Montreal infrastructure profiles. In general, the parks in the city have available parking lot infrastructure. Among much other analysis allowed by the graph, we can see the low correlation between the Water distribution dataset and the others, which system is under rebuilding in the last years. Besides the interconnection analysis among different services, we can observe the low interconnection results, for example, -0.09 in the criminal acts dataset. This result is due to good performance in the unsupervised phase that extracts each dataset's compact latent representation, excluding noise and mainly correlated features.

3.2.4.4 Discussion

The experiment's outcomes show that the ensemble learning model produces encouraging results for a wide range of data. Semi-optimal hyperparameters have been found in the training and validation phases, and used in the test phase. We could notice that our framework is suitable, and the stages kept the reconstructed errors at an adequate level considering the unsupervised

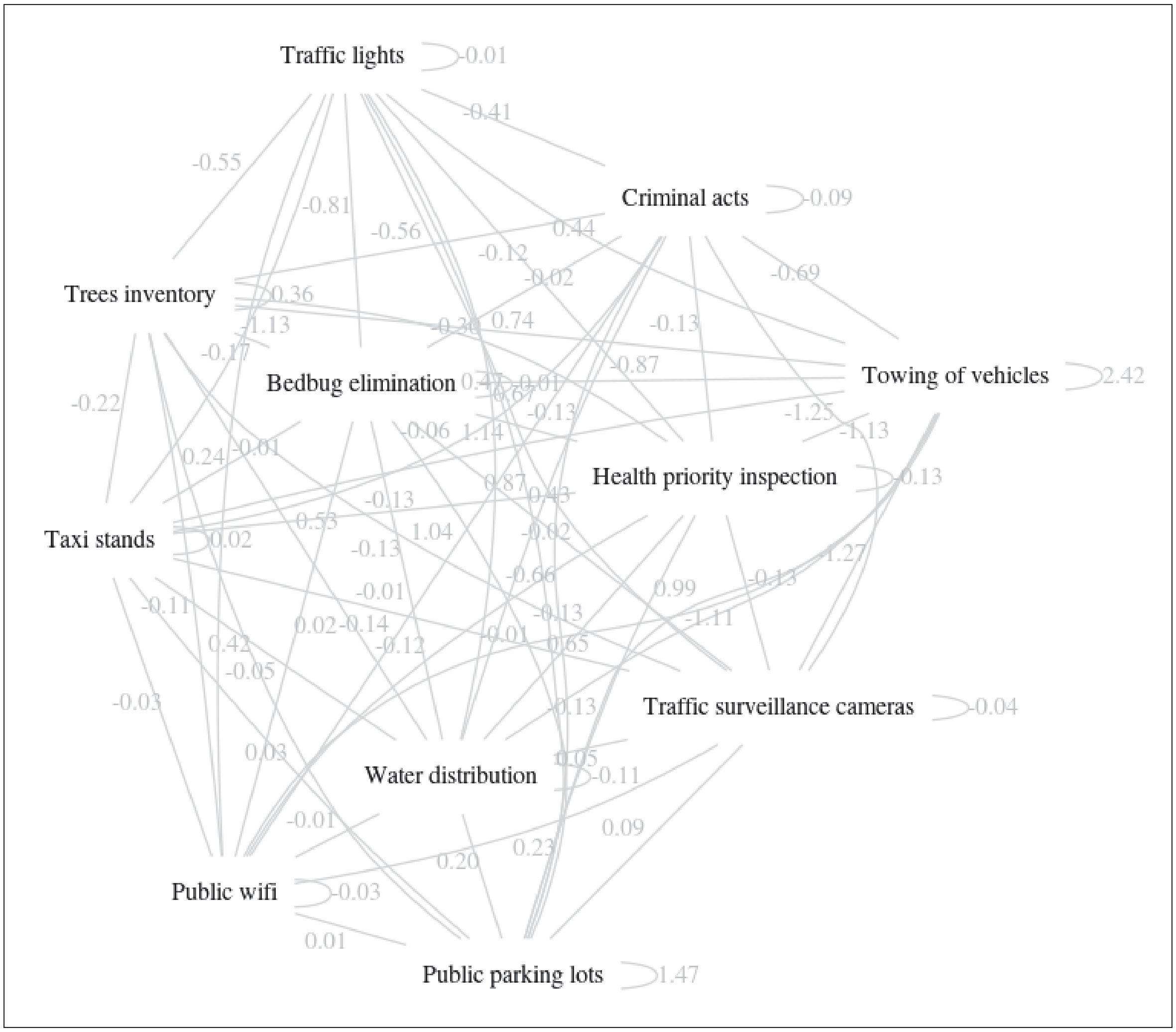


Figure 3.9 Graph of the fused services in Montreal

scenario, 0.135 in the test phase and 0.163 in the full fusion process. The proposed approach achieves its goal, avoiding overfitting and brightening up the relationship among the services provided by the city. In the literature, data fusion is highlighted as a time-consuming task, although there is a lack of such tools to support the whole process effectively. These results meet the requirements established in our specific objective **SO2**, showing that we effectively addressed the problem.

CONCLUSION AND RECOMMENDATIONS

In this thesis we attempted to address the very challenging problems of missing data imputation and data fusion in the smart environment context. A cross-domain analysis provides more efficient services. The data fusion process would allow a more complete, accurate and rapid evaluation of the cities' services. Also, the ability to fuse data enables context awareness which has huge potential to enhance the efficiency of the urban space management, and further can provide vital knowledge back to the citizens. Therefore, in this situation, data fusion means a multi-level process aimed to improve the awareness of the public or private spaces considering multiple sources of data.

Because of the high complexity related to heterogeneous and distributed datasets, its multiples sources, formats, languages and other items, we addressed two interconnected problems in this thesis. Firstly the missing data imputation, focusing on improve data quality, taking into consideration the drawbacks and harmfulness that it might bring to the fusion process itself and its outcomes. The second is the data fusion by fusing datasets from different domains such as transportation, health and environment among other, regarded to the geospatial semantic alignment. In order to solve the main challenge, we have broken it down into two parts and for each of them we have accomplished a comprehensive state-of-the-art proposing the most suitable machine learning models capable to cope the gap found in the literature review. Observing the quality of the data, specifically missing data, and how it is critical in the preprocessing step in data fusion, we addressed this problem separately. Our specific objective of tackling missing values was to enhance the power of the analysis and its quality results, keeping the statistical strength of the dataset, which might have this capacity degraded due to missing values. In the sequence, we designed a stacked ensemble learning framework to perform missing data imputation in heterogeneous datasets and then developed an ensemble deep learning framework for data fusion.

The methodology for missing values framework covers the widespread problem frequently found in real-world datasets, and it is able to deal with various data types, both labeled and unlabeled datasets. It comprises three phases: in the first, the RF produces a different subset of the data. In the second step, when applying the Jaccard index, we selected all subsets that match better with the subset where data is missing. Lastly, we used Bayes probability to assign the most probable value to impute the missing data. The second framework was able to tackle data fusion problem especially to cope with data heterogeneity. Our complete framework can indeed deal with feature in-feature out, Dasarathy classification through the semantic alignment principle. After proceeding with proper treatments in datasets, an RBM model has been employed to transform raw attributes in features that capture most of the latent representations of information. Meanwhile, the GloVe tool was used to tackle textual data due to the massive amount of this type of material in our case. Finally, we fed the DFMF framework to fuse the input features and receive the shared matrix factors, representing the whole structure.

The limitations of our missing data imputation framework, considering the future scenarios where a massive volume, variety, and velocity (big data pillars) of data are estimated are:

- the proper treatment of the velocity which was not the focus of this work.
- the implementation in the big data and ML platforms.

Our ensemble deep learning framework for data fusion intends to address fusion process in unlabeled datasets, the limitations of the framework are:

- the treatment of data privacy was not the focus of this work. However, currently, it is an important aspect of the information.
- in real-world datasets, only a small fraction is actually labelled. Then, if we label the datasets after our fusion process, it could be enriching information, allowing the data to be treated in a supervised fashion.

As for, the future directions that we can extend our missing data imputation framework, we are able to provide some solution to perform in online data to address the first limitation, considering that our ensemble learning framework for missing data imputation had already managed the missing values with respect to the volume and variety. To cover the second limitation in the feature, we can also work on its implementation in the big data and ML platforms such as Spark and R. Consequently, the fusion data framework may be extended in two directions: In first place, we can extend the framework to to ensure data security and data privacy in data fusion, in order to mitigate the third limitation. Secondly, labeling the fused data of our framework to perform supervised predictive tasks assuming that it is the primary reason in unsupervised machine learning, addressing the fourth limitation. Such extensions could contribute further.

APPENDIX I

ARTICLES PUBLISHED IN CONFERENCES

This thesis is related to two papers as follows:

1. “Ensemble learning for heterogeneous missing data imputation” Presented at 2020 International Conference on Big Data - September 18 - 20, 2020. Published in Proceedings LNCS 12402 Springer - Chapter 10. Carvalho *et al.* (2020)
2. “Ensemble Deep Learning Model for Data Fusion” Submitted to KDD 2021 14-18 August 2021 // Virtual conference.

Check for updates

Ensemble Learning for Heterogeneous Missing Data Imputation

Andre Luis Costa Carvalho(✉), Darine Ameyed(✉), and Mohamed Cheriet(✉)

System Engineering Department, University of Quebec's Ecole de Technologie Superieure, Montreal, QC H3C 1K3, Canada
{andre-luis.costa-carvalho.1,darine.ameyed.1}@ens.etsmtl.ca,
mohamed.cheriet@etsmtl.ca

Abstract. Missing values can significantly affect the result of analyses and decision making in any field. Two major approaches deal with this issue: statistical and model-based methods. While the former brings bias to the analyses, the latter is usually designed for limited and specific use cases. To overcome the limitations of the two methods, we present a stacked ensemble framework based on the integration of the adaptive random forest algorithm, the Jaccard index, and Bayesian probability. Considering the challenge that the heterogeneous and distributed data from multiple sources represents, we build a model in our use case, that supports different data types: continuous, discrete, categorical, and binary. The proposed model tackles missing data in a broad and comprehensive context of massive data sources and data formats. We evaluated our proposed framework extensively on five different datasets that contained labelled and unlabelled data. The experiments showed that our framework produces encouraging and competitive results when compared to statistical and model-based methods. Since the framework works for various datasets, it overcomes the model-based limitations that were found in the literature review.

Keywords: Ensemble methods · Missing data imputation · Distributed data · Multidomains · Big data · Smart city

Surya Nepal · Wenqi Cao ·
Aziz Nasridinov · MD Zakirul Alam Bhuiyan ·
Xuan Guo · Liang-Jie Zhang (Eds.)

LNCS 12402

Big Data –
BigData 2020

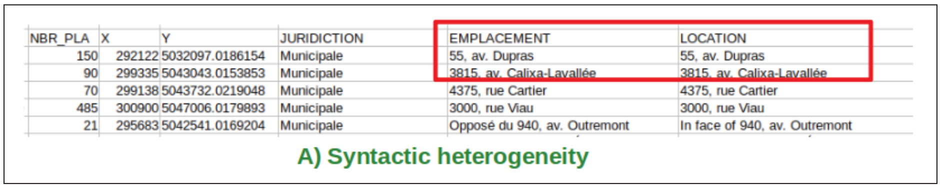
9th International Conference
Held as Part of the Services Conference Federation, SCF 2020
Honolulu, HI, USA, September 18–20, 2020
Proceedings

Big Data

Figure-A I-1 Published paper

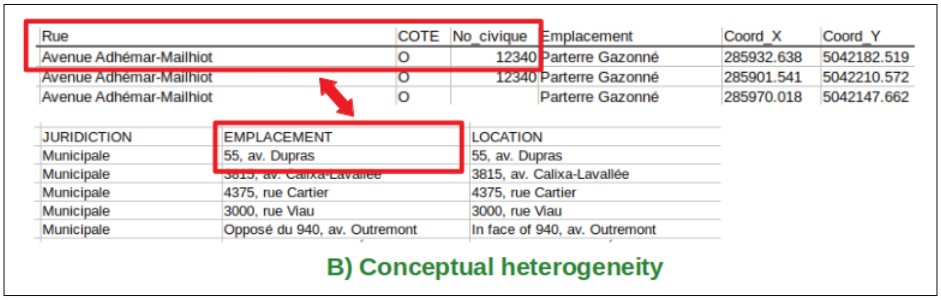
APPENDIX II

HETEROGENEOUS DATASETS EXAMPLES



NBR_PLA	X	Y	JURIDICTION	EMPLACEMENT	LOCATION
150	292122	5032097.0186154	Municipale	55, av. Dupras	55, av. Dupras
90	299335	5043043.0153853	Municipale	3815, av. Calixa-Lavallée	3815, av. Calixa-Lavallée
70	299138	5043732.0219048	Municipale	4375, rue Cartier	4375, rue Cartier
485	300900	5047006.0179893	Municipale	3000, rue Viau	3000, rue Viau
21	295683	5042541.0169204	Municipale	Opposé du 940, av. Outremont	In face of 940, av. Outremont

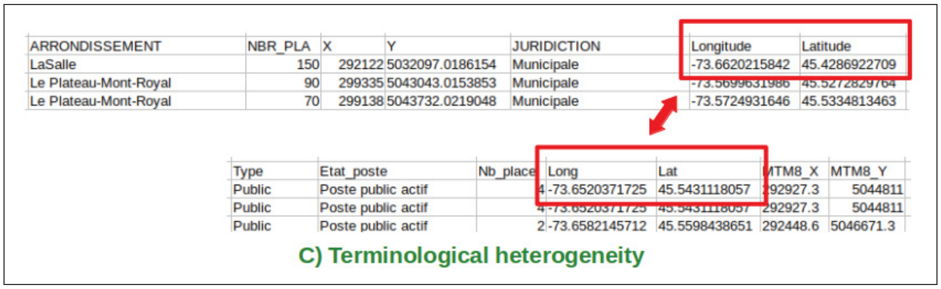
Figure-A II-1 Syntactic heterogeneity



Rue	COTE	No_civique	Emplacement	Coord_X	Coord_Y
Avenue Adhémar-Mailhiot	O	12340	Parterre Gazonné	285932.638	5042182.519
Avenue Adhémar-Mailhiot	O	12340	Parterre Gazonné	285901.541	5042210.572
Avenue Adhémar-Mailhiot	O		Parterre Gazonné	285970.018	5042147.662

JURIDICTION	EMPLACEMENT	LOCATION
Municipale	55, av. Dupras	55, av. Dupras
Municipale	3815, av. Calixa-Lavallée	3815, av. Calixa-Lavallée
Municipale	4375, rue Cartier	4375, rue Cartier
Municipale	3000, rue Viau	3000, rue Viau
Municipale	Opposé du 940, av. Outremont	In face of 940, av. Outremont

Figure-A II-2 Conceptual heterogeneity



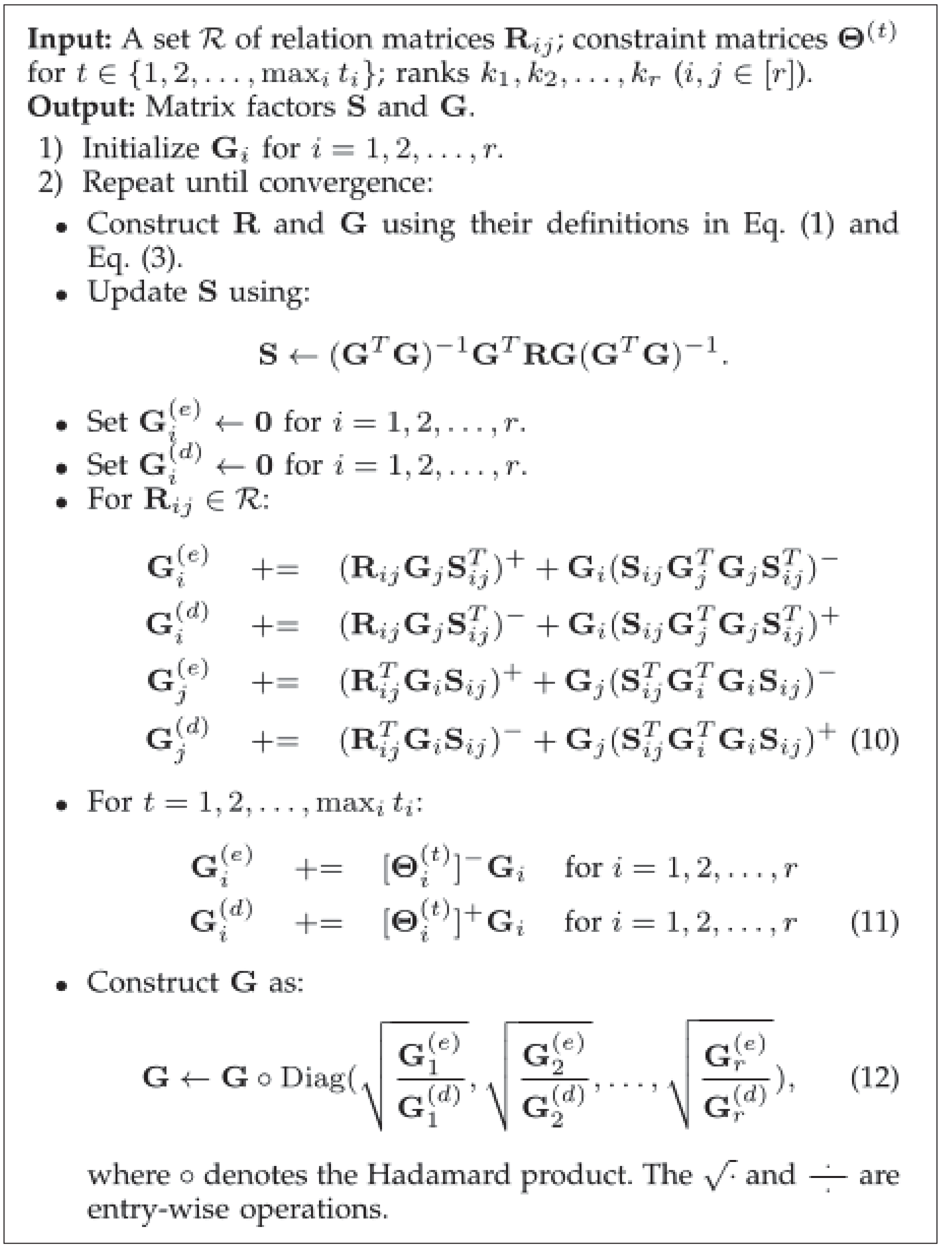
ARRONDISSEMENT	NBR_PLA	X	Y	JURIDICTION	Longitude	Latitude
LaSalle	150	292122	5032097.0186154	Municipale	-73.6620215842	45.4286922709
Le Plateau-Mont-Royal	90	299335	5043043.0153853	Municipale	-73.5699631986	45.5272829764
Le Plateau-Mont-Royal	70	299138	5043732.0219048	Municipale	-73.5724931646	45.5334813463

Type	Etat_poste	Nb_place	Long	Lat	MTM8_X	MTM8_Y
Public	Poste public actif	4	-73.6520371725	45.5431118057	292927.3	5044811
Public	Poste public actif	4	-73.6520371725	45.5431118057	292927.3	5044811
Public	Poste public actif	2	-73.6582145712	45.5598438651	292448.6	5046671.3

Figure-A II-3 Terminological heterogeneity

APPENDIX III

DFMF ALGORITHM



```
Input: A set $\mathcal{R}$ of relation matrices $\mathbf{R}_{ij}$; constraint matrices $\mathbf{\Theta}^{(t)}$ for $t \in \{1, 2, \ldots, \max_i t_i\}$; ranks $k_1, k_2, \ldots, k_r$ $(i, j \in [r])$.
Output: Matrix factors $\mathbf{S}$ and $\mathbf{G}$.
1) Initialize $\mathbf{G}_i$ for $i = 1, 2, \ldots, r$.
2) Repeat until convergence:
  • Construct $\mathbf{R}$ and $\mathbf{G}$ using their definitions in Eq. (1) and Eq. (3).
  • Update $\mathbf{S}$ using:
        $\mathbf{S} \leftarrow (\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T\mathbf{R}\mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}.$
  • Set $\mathbf{G}_i^{(e)} \leftarrow \mathbf{0}$ for $i = 1, 2, \ldots, r$.
  • Set $\mathbf{G}_i^{(d)} \leftarrow \mathbf{0}$ for $i = 1, 2, \ldots, r$.
  • For $\mathbf{R}_{ij} \in \mathcal{R}$:
        $\mathbf{G}_i^{(e)} \mathrel{+}= (\mathbf{R}_{ij}\mathbf{G}_j\mathbf{S}_{ij}^T)^+ + \mathbf{G}_i(\mathbf{S}_{ij}\mathbf{G}_j^T\mathbf{G}_j\mathbf{S}_{ij}^T)^-$
        $\mathbf{G}_i^{(d)} \mathrel{+}= (\mathbf{R}_{ij}\mathbf{G}_j\mathbf{S}_{ij}^T)^- + \mathbf{G}_i(\mathbf{S}_{ij}\mathbf{G}_j^T\mathbf{G}_j\mathbf{S}_{ij}^T)^+$
        $\mathbf{G}_j^{(e)} \mathrel{+}= (\mathbf{R}_{ij}^T\mathbf{G}_i\mathbf{S}_{ij})^+ + \mathbf{G}_j(\mathbf{S}_{ij}^T\mathbf{G}_i^T\mathbf{G}_i\mathbf{S}_{ij})^-$
        $\mathbf{G}_j^{(d)} \mathrel{+}= (\mathbf{R}_{ij}^T\mathbf{G}_i\mathbf{S}_{ij})^- + \mathbf{G}_j(\mathbf{S}_{ij}^T\mathbf{G}_i^T\mathbf{G}_i\mathbf{S}_{ij})^+$   (10)
  • For $t = 1, 2, \ldots, \max_i t_i$:
        $\mathbf{G}_i^{(e)} \mathrel{+}= [\mathbf{\Theta}_i^{(t)}]^-\mathbf{G}_i$   for $i = 1, 2, \ldots, r$
        $\mathbf{G}_i^{(d)} \mathrel{+}= [\mathbf{\Theta}_i^{(t)}]^+\mathbf{G}_i$   for $i = 1, 2, \ldots, r$   (11)
  • Construct $\mathbf{G}$ as:
        $\mathbf{G} \leftarrow \mathbf{G} \circ \mathrm{Diag}\left(\sqrt{\frac{\mathbf{G}_1^{(e)}}{\mathbf{G}_1^{(d)}}}, \sqrt{\frac{\mathbf{G}_2^{(e)}}{\mathbf{G}_2^{(d)}}}, \ldots, \sqrt{\frac{\mathbf{G}_r^{(e)}}{\mathbf{G}_r^{(d)}}}\right),$   (12)
    where $\circ$ denotes the Hadamard product. The $\sqrt{\cdot}$ and $\frac{\cdot}{\cdot}$ are entry-wise operations.
```

Figure-A III-1 Factorization algorithm of proposed data fusion approach:

BIBLIOGRAPHY

Aggarwal, C. C. & Srinivasan, P. (2001). “Mining Massively Incomplete Data Sets by Conceptual Reconstruction”. *Proceedings of the Seventh ACM SIGKDD*, 227–232. doi: doi: 10.1145/502512.502543.

Albergante, L., Mirkes, E. M., Chen, H., Martin, A., Faure, L., Barillot, E., Pinello, L., Gorban, A. N. & Zinovyev, A. (2018). Robust and scalable learning of data manifolds with complex topologies via ElPiGraph. *CoRR journal*. doi: [arxiv.org/abs/1804.07580].

Aldwairi, T., Perera, D. & A. Novotny, M. (2018). An evaluation of the performance of Restricted Boltzmann Machines as a model for anomaly network intrusion detection. *Computer Networks*, 144, 111-119. doi: https://doi.org/10.1016/j.comnet.2018.07.025.

Alice Zheng, A. C. (2018). *Feature Engineering for Machine Learning - Principles and Techniques for Data scientists*. O’reilly.

Allan, F. E. & Wishart, J. (1930). A Method of Estimating the Yield of a Missing Plot in Field Experimental Work. *The Journal of Agricultural Science*, 20(3), 399–406. doi: 10.1017/S0021859600006912.

Azimi, I., Pahikkala, T., Rahmani, A. M., Niela-Vilén, H., Axelin, A. & Liljeberg, P. (2019). Missing data resilient decision-making for healthcare IoT through personalization: A case study on maternal health. *Future Generation Computer Systems*, 96, 297–308. doi: https://doi.org/10.1016/j.future.2019.02.015.

Baraldi, P., Maio, F. D., Genini, D. & Zio, E. (2015). “Reconstruction of missing data in multidimensional time series by fuzzy similarity”. *Applied Soft Computing journal*, 26, 1–9. doi: "https://doi.org/10.1016/j.asoc.2014.09.038".

Bengio, Y., Courville, A. & Vincent, P. (2012). Representation Learning: A Review and New Perspectives.

Bishop, C. (2013). “Model-based machine learning”. *Philosophical Transactions of the Royal Society*. doi: doi: https://doi.org/10.1098/rsta.2012.0222.

Botega, L. C., Junior, V. A. P., Pereira, G. M. C., de Oliveira, A. C. M., Saran, J. F., Ladeira, L. Z. & Isotani, S. (2019). Quantify: An Information Fusion Model Based on Syntactic and Semantic Analysis and Quality Assessments to Enhance Situation Awareness. *Information Quality in Information Fusion and Decision Making*, 563-586. doi: 10.1007/978-3-030-03643-0_23.

Bubenik, P. (2012). Statistical topological data analysis using persistence landscapes.

Carvalho, A. L. C., Ameyed, D. & Cheriet, M. (2020). Ensemble Learning for Heterogeneous Missing Data Imputation. *Big Data – BigData 2020*, pp. 127–143.

Castanedo, F. (2013). A Review of Data Fusion Techniques. *The Scientific World Journal*, 19. doi: 10.1155/2013/704504.

Cearly, D. W. (2019). *Top 10 strategic technology trends for 2019*. Gartner Inc. and/or its affiliates. All rights reserved. PR575107.

Cordova, I. & Moh, T. (2015). DBSCAN on Resilient Distributed Datasets. *2015 International Conference on High Performance Computing Simulation (HPCS)*, pp. 531-540. doi: 10.1109/HPCSim.2015.7237086.

Crowley, J. L. (1984). A Computational Paradigm for 3-D Scene Analysis. *IEEE Conf. on Computer Vision, Representation and Control*.

Dasarathy, B. V. (1997). Sensor fusion potential exploitation - Innovative Architectures and Illustrative Applications. *Proceedings of the IEEE*, 85(1), 24-38. doi: 0018-9219(97)00642-7.

de Oliveira, V. & Kedem, B. (2017). *Statistical Data Fusion*. World Scientific Publishing Co. Pte. Lyd.

de Oliveira, V. & Kedem, B. (2018). Bayesian analysis of a density ratio model. 45, 274–289.

Dean, J. & Ghemawat, S. (2008). MapReduce: Simplified Data Processing on Large Clusters. *Commun. ACM*, 51(1), 107–113. doi: 10.1145/1327452.1327492.

Department of Economic and Social Affairs. (2019). *World Urban Prospects The 2018 Revision*. United Nations.

Di Zio, M., Fursova, N. & T., G. (2016). *Methodology for data validation 1.0*. Essnet Validat Foundation. Consulted at https://ec.europa.eu/eurostat/cros/system/files/methodology_for_data_validation_v1.0_rev-2016-06_final.pdf.

Durrant-Whyte & Hugh, F. (1990). Sensor Models and Multisensor Integration. In Cox, I. J. & Wilfong, G. T. (Eds.), *Autonomous Robot Vehicles* (pp. 73–89). New York, NY: Springer New York. doi: 10.1007/978-1-4613-8997-2_7.

Durrant-Whyte, H. F. (1987). Consistent Integration and Propagation of Disparate Sensor Observations. *The International Journal of Robotics Research*, 6(3), 3-24. doi: 10.1177/027836498700600301.

Essien, A., Petrounias, I., Sampaio, P. & Sampaio, S. (2019). Improving Urban Traffic Speed Prediction Using Data Source Fusion and Deep Learning. *2019 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 1-8.

Faugeras, O., Ayache, N. & Faverjon, B. (1986). Building visual maps by combining noisy stereo measurements. *Proceedings. 1986 IEEE International Conference on Robotics and Automation*, 3, 1433-1438. doi: 10.1109/ROBOT.1986.1087419.

Ferrer, J. R. (2017). Barcelona's Smart City vision: an opportunity for transformation. *Field Actions Science Reposts - The journal of field action*.

Geerts, F., Mecca, G., Papotti, P. & Santoro, D. (2019). "Cleaning data with Llunatic". *The VLDB Journal*. doi: doi: 10.1007/s00778-019-00586-5, 2019.

Graham, J. W. (2012). *Missing Data - Analysis and Design*. Springer-Verlag New York. doi: 10.1007/978-1-4614-4018-5.

Gudivada, V. N., Rao, D. & Raghavan, V. V. (2015). "Big Data Driven Natural Language Processing Research and Applications". "33", 203–238. doi: https://doi.org/10.1016/B978-0-444-63492-4.00009-5.

Hall, D. & Llinas, J. (1997). An Introduction to Multisensor Data Fusion. *Proceedings of the IEEE*, 85, 6 - 23. doi: 10.1109/5.554205.

Hatem, B. S. (2017). Quality and the efficiency of data in "Smart-Cities". *Future Generation Computer Systems*, 74, 409–416. doi: https://doi.org/10.1016/j.future.2016.12.021.

Hinton, G. (2012). A Practical Guide to Training Restricted Boltzmann Machines. *Neural Networks: Tricks of the Trade*, 7700. doi: https://doi.org/10.1007/978-3-642-35289-8_32.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8), 2554–2558. doi: 10.1073/pnas.79.8.2554.

Housfater, A. S., Zhang, X. P. & Zhou, Y. (2006, May). Nonlinear Fusion of Multiple Sensors with Missing Data. *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, 4, IV-IV. doi: 10.1109/ICASSP.2006.1661130.

Jegelka, S., Kapoor, A. & Horvitz, E. (2013). An Interactive Approach to Solving Correspondence Problems. *International Journal of Computer Vision*, 108, 49-58. doi: 10.1007/s11263-013-0657-5.

Kuhn, H. W. (2006). The Hungarian method for the assignment problem. *Wiley Periodicals, Inc., A Wiley Companyl*. doi: https://doi.org/10.1002/nav.3800020109.

Lau, B. P. L., Marakkalage, S. H., Zhou, Y., Hassan, N. U., Yuen, C., Zhang, M. & Tan, U.-X. (2019). A survey of data fusion in smart city applications. *Information Fusion*, 52, 357 - 374. doi: https://doi.org/10.1016/j.inffus.2019.05.004.

Little Roderick, J. A. & Rubin, D. B. (2002). *Statistical Analysis with Missing Data, Second Edition* (ed. 2). Wiley Series in Probability and Statistics.

Marjani, M., Nasaruddin, F., Gani, A., Karim, A., Hashem, I. A. T., Siddiqa, A. & Yaqoob, I. (2017). Big IoT Data Analytics: Architecture, Opportunities, and Open Research Challenges. *IEEE Access*, 5, 5247-5261. doi: 10.1109/ACCESS.2017.2689040.

Mayo-Wilson, E., Li, T., Fusco, N. & Dickersin, K. (2018). Practical guidance for using multiple data sources in systematic reviews and meta-analyses. *MUDS investigators*, 9, 2–12. doi: doi:10.1002/jrsm.1277.

Meng, T., Jing, X., Yan, Z. & Pedrycz, W. (2020). A survey on machine learning for data fusion. *Information Fusion*, 57, 115–129. doi: 10.1109/TPAMI.2014.2343973.

Mitchell, H. B. (2010). *Data Fusion: Concepts and Ideas*. Springer. doi: DOI 10.1007/978-3-642-27222-6.

Mnih, A. & Kavukcuoglu, K. (2013). Learning Word Embeddings Efficiently with Noise-Contrastive Estimation. *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, (NIPS'13), 2265–2273.

Mohamed, A. & Hinton, G. (2009). Deep belief networks for phone recognition. *In NIPS 22 workshop on deep learning for speech recognition*, pp. 1-8.

Mohan, K. & Pearl, J. (2018). Graphical Models for Processing Missing Data. doi: 978-1-61284-385-8.

Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. The MIT Press.

Musil, C. M., Warner, C. B., Yobas, P. K. & Jones, S. L. (2002). A comparison of imputation techniques for handling missing data. *Western Journal of Nursing Research*, 24(7), 815–829. doi: 10.1177/019394502762477004.

Pavlidis, P., Weston, J., Cai, J. & Noble, W. S. (2002). Learning gene functional classifications from multiple data types. *J Comput Biol*, 9, 401–411. doi: doi:10.1089/10665270252935539.

Pennington, J., Socher, R. & Manning, C. (2014). GloVe: Global Vectors for Word Representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543. doi: 10.3115/v1/D14-1162.

Petrozziello, A., Jordanov, I. & Sommeregger, C. (2018). "Distributed Neural Networks for Missing Big Data Imputation". *International Joint Conference on Neural Networks (IJCNN)*, 1–8. doi: doi: 10.1109/IJCNN.2018.8489488.

Qin, X. & Gu, Y. (2011). Data fusion in the Internet of Things. *Procedia Engineering*, 15, 3023-3026. doi: 10.1016/j.proeng.2011.08.567.

Schafer, L. & Graham, J. W. (2002). Missing Data: Our View of the State of the Art. *Psychological Methods journal*, 7, 147–177. doi: http://dx.doi.org/10.1037/1082-989X.7.2.147.

Schuster, C. & Vitoux, R. R. (2018). Methodology for Ensuring Accuracy and Validity of Infusion Pump Alarm Data. *Biomedical Instrumentation Technology*, 52(3), 192-198. doi: 10.2345/0899-8205-52.3.192.

Shahat, A., Elragal, A. & Bergvall-Kareborn, B. (2017). Big Data Analytics and Smart Cities: A Loose or Tight Couple?

Sta, H. B. (2017). Quality and the efficiency of data in "Smart-Cities". *Future Generation Computer Systems*, 74, 409-416. doi: https://doi.org/10.1016/j.future.2016.12.021.

Sun, B. & Saenko, K. (2015). Correlation Alignment for Deep Domain Adaptation.

Sun, B., Feng, J. & Saenko, K. (2016). Correlation Alignment for Unsupervised Domain Adaptation.

Tan, Y., Zhang, C., Mao, Y. & Qian, G. (2015, June). Semantic presentation and fusion framework of unstructured data in smart cites. *2015 IEEE 10th Conference on Industrial Electronics and Applications (ICIEA)*, pp. 897–901. doi: 10.1109/ICIEA.2015.7334237.

Udell, M., Horn, C., Zadeh, R. & Boyd, S. (2016). "Generalized Low Rank Models". *Foundations and Trends in Machine Learning*, 9(1), 1-118. doi: 10.1561/2200000055.

Vasudevan, S., Ramos, F., Nettleton, E. & Durrant-Whyte, H. (2011). Non-stationary dependent Gaussian processes for data fusion in large-scale terrain modeling. doi: 978-1-61284-385-8.

Wang, L. (2017). Heterogeneous Data and Big Data Analytics. *Automatic Control and Information Sciences*, 3(1), 8–15. doi: 10.12691/acis-3-1-3.

Wang, Y., Yu, G., Domeniconi, C., Wang, J., Zhang, X. & Guo, M. (2019). Selective Matrix Factorization for Multi-relational Data Fusion. *Database Systems for Advanced Applications*, pp. 313–329.

Wielenga, D. (2007). Identifying and Overcoming Common Data Mining Mistakes.

Zhou, Z. H. (2012). *Ensemble Methods: Foundations and Algorithms* (ed. 1). Boca Raton, FL, USA: Chapman Hall/CRC.

Zitnik, M. & Zupan, B. (2015). Data Fusion by Matrix Factorization. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 37. doi: 10.1109/TPAMI.2014.2343973.