Extraction automatique de contenu sur des forums hébergeant des communautés criminelles

par

Olivier MICHAUD

MÉMOIRE PRÉSENTÉ À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
COMME EXIGENCE PARTIELLE À L'OBTENTION DE LA MAÎTRISE
AVEC MÉMOIRE
M. Sc. A.

MONTRÉAL, LE "24 MARS 2022"

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

PRÉSENTATION DU JURY

CE MÉMOIRE A ÉTÉ ÉVALUÉ

PAR UN JURY COMPOSÉ DE:

Mme. Sylvie Ratté, Directrice de mémoire
Génie logiciel et des TI, École de technologie supérieure

M. Pierre André Ménard, co-Directeur
Traitement automatique des langues naturelles, Centre de recherche informatique de Montréal

M. Luc Duong, Président du jury
Génie logiciel et des TI, École de technologie supérieure

M. Patrick Cardinal, Membre du jury
Génie logiciel et des TI, École de technologie supérieure

IL A FAIT L'OBJET D'UNE SOUTENANCE DEVANT JURY ET PUBLIC

LE "15 MARS 2022"

À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

REMERCIEMENTS

Je tiens tout d'abord à remercier la directrice de ce mémoire, Madame Sylvie Ratté, qui a accepté de me prendre comme étudiant lorsque je complétais mon baccalauréat. Elle a su me guider à travers toutes les étapes menant à la réalisation de ce mémoire. Elle m'a consacré son temps précieux durant toutes ces années. Avec elle, j'ai grandi, j'ai ri, mais surtout, je me suis surpassé.

Je veux aussi exprimer mon plus grand respect pour Pierre André Ménard, co-Directeur de ce mémoire. Pierre André s'est impliqué dès le premier jour dans ce projet. Ses conseils, ses critiques et sa patience ont permis d'amener ce mémoire à un tout autre niveau.

Un grand merci à l'entreprise Flare Systems, et tous mes collègues, de m'avoir donné ma chance pour la réalisation de ce mémoire. Très rapidement, ils m'ont permis d'accéder à leurs données ainsi qu'à leurs connaissances. Je suis fièr du travail accompli avec eux, de la contribution apportée par ce travail au sein de leur plateforme. Mention spéciale à Simon Landry-Pellerin pour ses conseils et son implication dans le projet.

Je remercie aussi Alexandre Viau de m'avoir présenté cette opportunité au sein de la compagnie Flare Systems. Alexandre m'a appris de nombreuses choses, et j'en ressors un meilleur ingénieur logiciel grâce à lui.

Finalement, je tiens à remercier ma famille et mes amis pour leur soutien et leurs encouragements lors de la réalisation de ce mémoire.

Extraction automatique de contenu sur des forums hébergeant des communautés criminelles

Olivier MICHAUD

RÉSUMÉ

Autrefois isolé par les frontières géographiques, les communautés criminelles profitent de l'anonymat fourni par certains réseaux, tel le réseau TOR, faisant partie de l'Internet clandestin (Dark Web), afin de coopérer, vendre et partager leurs connaissances. Pour une organisation, analyser ces échanges sur les multiples forums présents sur ce réseau permet de détecter les tendances et ainsi prévenir de futures attaques. Par conséquent, ce mémoire propose une nouvelle approche afin de généraliser l'extraction des sujets de discussions et ses attributs (titre, auteur et date de publication). Celle-ci est portée par l'hypothèse qu'il est possible d'utiliser des outils provenant du traitement automatique des langues naturelles (TALN) afin de procéder à l'extraction de contenu sur le Web.

Afin de procéder à l'extraction des sujets de discussions, deux sous-objectif sont poursuivis. Le premier consiste à utiliser des outils d'annotation de séquences afin d'identifier les enregistrements et leurs attributs dans une page Web. Le deuxième est de procéder à l'extraction du contenu identifié. Pour y parvenir, une méthode est définie afin de transformer une page Web en une séquence composée de balises HTML et de texte. Il est alors possible de procéder à l'annotation de séquences sur celle-ci avec un modèle BiLSTM-CRF. La séquence est ensuite reconstruite en page Web afin de procéder à l'extraction des sujets de discussions. Pour ce faire, des algorithmes d'extraction ont été conçus, tirant avantage de la structure en graphe des pages HTML.

Suite aux expériences menées, qui consistaient à déterminer les meilleurs hyperparamètres et tailles de vocabulaire pour le modèle, il est possible de confirmer l'hypothèse de ce mémoire. En effet, les bons résultats sur le jeu de de tests de l'ensemble A (macro F1 de 99,5%), ainsi que les performances en contexte industriel, démontrent que la solution développée a su généraliser la structure des forums. Par conséquent, il est possible d'extraire des sujets de discussions sur de nouveaux forums qui n'ont pas été utilisés lors du processus d'entraînement du modèle.

Mots-clés: moissonnage du Web, traitements automatique des langues naturelles, annotation de séquences, forums clandestins

Titre en anglais

Olivier MICHAUD

ABSTRACT

Once isolated by geographical borders, criminal communities now take advantage of the anonymity provided by certain networks, such as the TOR network, part of the Dark Web, to cooperate, sell and share their knowledge. For an organization, analyzing these exchanges on the multiple forums present on this network allows detecting trends and thus preventing future attacks. Therefore, this thesis proposes a new approach to generalize the extraction of forum topics and their attributes (title, author and publication date). This approach is based on the hypothesis that it is possible to perform content extraction on the Web using natural language processing (NLP) tools.

In order to extract forum topics, two sub-objectives are defined. The first one is to use sequence labeling to identify records and their attributes on Web pages. The second one is to proceed to the extraction of the identified content. To achieve this, a method is defined to transform a Web page into a sequence composed of HTML tags and text. It is then possible to proceed to sequence labeling using a BiLSTM-CRF model. The sequence is then reconstructed into a Web page in order to proceed to the extraction of the forum topics. For this purpose, extraction algorithms have been designed, taking advantage of HTML page's graph structure.

Following the experiments (hyperparameters tuning, vocabulary size adjustment) it is possible to confirm the hypothesis of this thesis. Indeed, the good results on the test set (macro F1 of 99,5 %), as well as the performances in industrial context, demonstrate that the proposed solution was able to generalize the structure of the forums. Consequently, it is possible to extract forum topics from forums that were not used during the training process.

Keywords: underground Forums, web Scraping, natural language processing, sequence Labeling

TABLE DES MATIÈRES

Page

LISTE DES TABLEAUX

LISTE DES FIGURES

Page

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

BPE	Byte-Pair Encoding
CB	Nombre de couches de BiLSTM
CSS	Cascading Style Sheets
DA	Décroissance de poids
DOM	Document Object Model
DP	Dimension du plongement de mots
HD	Nombre de cellules LSTM
HTML	The HyperText Markup Language
TA	Taux d'apprentissage
TALN	Traitement automatique des langues naturelles
TOR	The Onion Router,

INTRODUCTION

Avec l'émergence de nouveaux canaux de communication assurant l'anonymat ainsi que la montée en popularité des cryptomonnaies, de plus en plus de communautés d'acteurs malicieux se sont formées sur le Web (Blanc, Hétu & Lavoie, 2021). Ces acteurs s'échangent des techniques criminelles en constante évolution, telles que des méthode de rançonnage, de vol de données et de vol d'identité, sur de nombreux forums et places de marchés (Guccione, 2021). Ces plateformes sont en grande partie hébergées sur le réseau TOR qui fait partie du canal de communication qu'est l'Internet clandestin (Dark Web).

L'anonymat que procure ce réseau rend la tâche de protection plus complexe pour les autorités et les entreprises touchées par ces attaques. Les acteurs criminels sont de plus en plus imprévisibles, en raison de l'amélioration de leurs techniques, coordination (amélioré par les réseaux anonymes) et leurs procédures (Abu, Rahayu, Ariffin (DrAA) & Robiah, 2018). De ce fait, selon un sondage effectué par le groupe CyberEdge, 86% des compagnies de plus de 500 employés ont été victimes d'une cyberattaque réussie en 2020 (CyberEdge Group, 2020). Par ailleurs, 60% des annonces placées sur l'Internet clandestin (excluant le trafic de marchandise illégale) sont susceptibles d'impacter une entreprise (Guccione, 2021). Un exemple de ces attaques est le logiciel de ranconnage WannaCry qui a affecté plus de deux millions d'ordinateurs dans plus de 150 pays (Mohurle & Patil, 2017). Celui-ci a affecté des ordinateurs d'institutions gouvernementales, d'universités, de compagnies et d'hôpitaux.

Pour une compagnie, la surveillance de ces plateformes fait partie d'une stratégie globale de renseignements sur les menaces (threat intelligence). Cette stratégie permet d'examiner des attaques passées et évaluer les futurs angles d'attaques afin de préparer une réponse efficace (Clay, 2021). Parmi les renseignements pertinents à récolter lors de cette surveillance des plateformes, on retrouve entre autres des vulnérabilités logiciels, des accès à des systèmes, des combinaisons de nom d'utilisateur et mot de passe (qui auront fuité ou bien été volés). (Den Berg, 2020).

Afin d'automatiser le processus de surveillance, des technologies des domaines de la surveillance du Web (Web Crawling) et moissonnage du Web (Web Scraping) sont utilisées. Le domaine de la surveillance du Web fait référence aux processus qui consiste à parcourir le Web automatiquement à l'aide d'un robot afin d'indexer son contenu. Le moissonnage du Web, quant à lui, est l'ensemble des techniques permettant d'extraire du contenu d'une page HTML. Ces technologies reposent principalement sur des configurations manuelles, ce qui augmente les coûts d'utilisation.

Dans ce mémoire, une méthode est proposée afin d'automatiser l'extraction de données sur des forums utilisés par les communautés de cybercriminels. Plus précisément, l'accent sera mis sur les pages contenant des sujets de discussions, desquels seront extraits le titre, l'auteur ainsi que la date de publication, pour chacune des entrées. La méthode proposée permet d'adapter une page HTML afin de procéder à de l'annotation de séquences, soit une technique du domaine du traitement automatique des langues naturelles (TALN). Les données utilisées sont fournies par l'entreprise Flare Systems.

CHAPITRE 1

ÉNONCÉ DU PROBLÈME

Les plateformes hébergeant le contenu des communautés de cybercriminels, principalement des forums et des places de marchés, ont proliféré au cours des dernières années, multipliant les incidents liés à sécurité pour plusieurs entreprises. De ce fait, les coûts estimés liés à la cybercriminalité ont atteint les 600 milliards en devise américaine en 2021, comparativement à 300 milliards en 2015 (Morgan, 2019). Ces coûts comprennent entre autres la perte et la destruction de données, la perte de productivité ainsi que le vol de propriété intellectuelle (Morgan, 2020).

Plusieurs de ces attaques ont été réalisées à la suite d'informations communiqués sur l'Internet clandestin, qui est un canal de communication dont fait partie le réseau TOR (Blanc *et al.*, 2021). Le réseau TOR permet un échange anonyme entre ses participants par le biais d'une série de serveurs mandataires (proxy) qui redistribue de façon anonyme les requêtes des usagers. On y trouve principalement des forums et places de marchés peuplés par des acteurs criminels attirés par l'anonymat qui y est offert (Guccione, 2021). Ces lieux sont des lieux d'échange d'outils frauduleux, tel des logiciels de rançonnage ou bien de vol d'identité, en plus de la vente d'accès systèmes ou bien de renseignement d'usagers (Den Berg, 2020).

Les pertes de données, de réputation et de productivité engendrées par ces attaques encouragent les compagnies à se prévenir contre celles-ci. Afin d'y arriver, une approche proactive consistant à surveiller les plateformes de discussions est de plus en plus envisagée. C'est ce que propose l'entreprise Flare Systems, partenaire de ce projet de recherche, qui surveille chaque jour en temps réel les plateformes de discussions utilisées par les cybercriminels.

Dans les prochaines sections, la contribution de ce travail de recherche est présentée (section 1.1). Par la suite, l'importante de l'automatisation des outils de surveillance des communautés

de cybercriminelles est introduite (section 1.2), suivi par l'introduction aux problématiques qu'implique l'extraction automatique sur le Web (section 1.3), et plus précisément, sur des plateformes de discussions de type forum (section 1.4). Finalement, l'hypothèse de recherche est présentée (section 1.5) ainsi que les objectifs (section 1.6).

1.1 Contribution de ce mémoire

Dans ce mémoire, une nouvelle approche est présentée afin de procéder à de l'extraction automatique de contenu sur le Web. Cette approche est basée sur l'hypothèse qu'il est possible d'utiliser des techniques provenant du domaine du TALN afin de détecter et extraire les attributs présents dans une page Web. De ce fait, une méthodologie est proposée afin d'adapter les pages Web en une séquence unique composées de balises HTML et de texte et d'utiliser des outils d'annotation de séquences sur celle-ci. La solution finale est en mesure de généraliser l'extraction des attributs de sujets de discussions (titre, auteur et date de publication), et ce, sans intervention humaine. De plus, celle-ci est en mesure de procéder à l'extraction des sujets de discussions sur les sites Web utilisés lors de l'entrainement, mais aussi sur de nouveaux sites Web, qui sont inconnus au modèle.

1.2 Automatisation des outils de surveillance des communautés de cybercriminelles en ligne

Surveiller les plateformes hébergeant des communautés de cybercriminels est l'une des cinq stratégies prônées par le groupe CyberEdge afin de protéger des entreprises contre des cyberattaques (CyberEdge Group, 2020). En effet, connaître les tendances du moment permet d'optimiser les ressources disponibles et ainsi diminuer le temps de réaction. Cependant, réaliser cette tâche manuellement requiert des spécialistes qui doivent naviguer parmi les plateformes et identifier les menaces. Avec l'augmentation constante du trafic sur ces plateformes et la pénurie

de main-d'œuvre dans le secteur, l'automatisation de ces tâches devient nécessaire (Spielman, 2016).

Afin de réduire la quantité de données à traiter manuellement, des efforts ont été déployés dans les dernières années afin de filtrer et classifier les publications se trouvant sur ces plateformes (Pastrana, Thomas, Hutchings & Clayton, 2018). Cependant, ces solutions sont grandement limités par la complexité de l'accès aux données pertinentes. En effet, les données utilisées dans ce type de recherche proviennent principalement de partenaires de recherche (Kadoguchi, Hayashi, Hashimoto & Otsuka, 2019) ou bien sont acquises avec la mise en place d'un système personnalisé de surveillance du Web et de moissonnage du Web (Nunes *et al.*, 2016; Macdonald, Frank, Mei & Monk, 2015) (voir section 1.3). De ce fait, Kadoguchi *et al.* (2019) associent leurs résultats décevants sur les données non vues durant l'entraînement (précision 90% sur les données d'entraînement, 79,4% sur les données de test) au manque de données. De plus, Nunes *et al.* (2016) ont démontré que l'ajout de données non libellées à des fins d'apprentissage non supervisé augmentait les performances de classification (précision de 80% et rappel de 78%) comparativement à une approche strictement supervisée (précision de 80% et rappel de 68%).

Un accès facile à une plus grande masse de données sur les acteurs malicieux passe donc par une acquisition automatisée. Cette automatisation rencontre cependant de nombreux défis :

- Tout d'abord, les données extraites doivent être structurées. En effet, il est important de pouvoir établir un lien entre un acteur et ses publications. De ce fait, Nunes *et al.* (2016) notent qu'il est fréquent que les vendeurs procèdent à la promotion de leurs produits dans des forums pour ensuite les offrir dans des places de marchés (Nunes *et al.*, 2016).
- De plus, la surveillance doit se faire en temps quasi réel afin de détecter les attaques de type "vulnérabilité du jour zéro" le plus rapidement possible. Ces attaques sont inconnues des éditeurs de logiciels et n'ont donc pas de correctif connu.

- Finalement, de nouvelles plateformes émergent chaque jour, tandis que d'autres cessent leurs activités. Il faut alors constamment être à l'affût des tendances.

1.3 Extraction automatique de contenu sur le Web

Surveiller une source, soit un ensemble de pages Web sous un même domaine, demande plusieurs champ d'expertise. Des techniques d'exploration du Web permettent de configurer des robots qui vont parcourir les pages Web pertinentes. Les informations contenues dans ces pages seront par la suite extraites avec l'aide d'outils faisant partie du domaine du moissonnage de Web. Ces informations doivent alors être structurées et ajoutées dans une base de données. Ces deux domaines, soient l'exploration du Web et le moissonnage du Web, représentent chacun d'eux un domaine de recherche en soit. Ce mémoire se concentrera sur le moissonnage du Web.

Le moissonnage du Web, tel que défini par Singrodia, Mitra & Paul (2019), est la procédure d'extraction automatique de l'information sur une page Web sans la copier manuellement. En effet, les données présentes sur une page Web sont principalement conçues pour être affichées dans un navigateur. Celles-ci ne sont pas accessibles sous un format structuré à des fins de téléchargements personnels. Elles sont encapsulées dans une mise en page prédéfinie qui peut changer d'un site Web à l'autre ou à travers le temps.

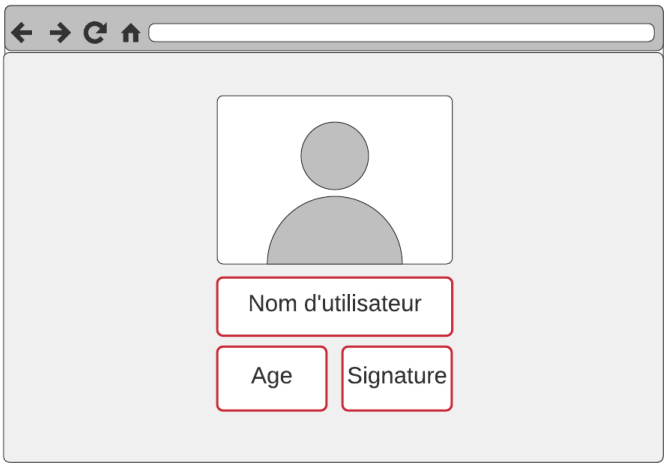
Les défis d'implémenter une solution d'extraction de contenu sur le Web sont nombreux, tels qu'énumérés dans Ferrara, De Meo, Fiumara & Baumgartner (2014) :

- Les approches d'extraction de contenu sur le Web demandent souvent l'aide d'experts humains afin de configurer et maintenir les outils d'extraction. Pour une entreprise comme Flare Systems, le défi est de réduire l'apport de ces experts afin d'obtenir un meilleur niveau d'automatisation, tout en gardant un grand niveau de précision.

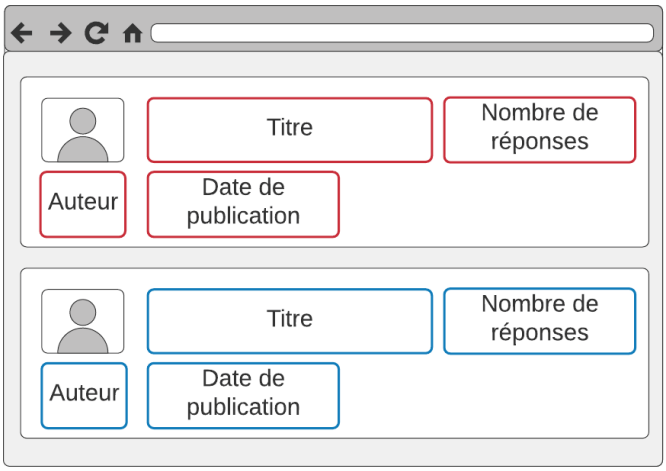
- Les techniques d'extraction doivent être en mesure de traiter une grande quantité de pages, et par conséquent, de types de données, en peu de temps. Ceci est particulièrement vrai dans le domaine de la surveillance de communautés d'acteurs malicieux tel que spécifié dans la section 1.2.
- Les aspects social et éthique doivent être pris en compte. Les données privées des utilisateurs doivent être traitées selon les politiques de chaque plateforme. Dans le cas de la surveillance des acteurs malveillants, Pastrana *et al.* (2018) résume les enjeux éthiques qui y sont reliés. Parmi ceux-ci, on retrouve le débat sur ce qui est public sur le Web.
- Les approches qui reposent sur l'apprentissage machine demandent une bonne quantité de données d'entraînement libellées. Ces données peuvent être longues à acquérir, particulièrement si elles sont annotées manuellement. Dans le cadre de ce mémoire, il n'existe en effet aucune base de données connue ou des sujets de discussion sont annotés au sein de pages Web provenant de forums de discussions clandestins.
- La nature des pages Web fait en sorte qu'elles évoluent, que ce soit suite à un changement de thème ou bien une mise à jour. Ces changements peuvent impacter les outils d'extraction et les amener à cesser de fonctionner correctement. Dans le cas de la compagnie Flare Systems, ces changements font en sorte qu'un expert humain doit continuellement surveiller le bon fonctionnement des outils d'extractions (voir section 2.1.1)

En plus de ces défis, il est important de s'attarder aux deux types de pages à traiter qui sont présentes sur le Web. La première est la page Web qui contient un seul enregistrement (record), tel qu'illustré à la figure 1.1a. Un enregistrement étant une entité contenant des attributs (tel un titre dans un sujet de discussion). Lors du processus d'extraction, les attributs seront extraits de la page et associés à cet enregistrement. Le deuxième type de page Web est illustré à la figure 1.1b, et représente une page Web contenant plusieurs enregistrements. Chacun de ses

enregistrements contient ses propres attributs. Cette propriété doit être prise en compte lors du processus d'extraction.



a) Une page Web avec un seul enregistrement



b) Une page Web avec plusieurs enregistrements

Figure 1.1 Représentation de deux types de pages Web. Dans chaque exemple, les attributs appartenant au même enregistrement sont illustrés dans la même couleur

Afin d'extraire automatiquement les enregistrements des pages Web, les compagnies peuvent se tourner vers quatre types de solutions, telles que résumées dans Dilmegani (2020) :

- Les solutions provenant des logiciels libres (open source), qui contiennent des librairies permettant de configurer manuellement l'extraction de contenu provenant du Web.
- Les solutions propriétaires, qui permettent d'héberger un logiciel configurable permettant de parcourir le Web et d'extraire des données.
- Les logiciels en tant que service, qui sont fournis par des compagnies spécialisées en extraction de contenu sur le Web. Celles-ci offrent leur service afin que les clients aient la possibilité de configurer le système selon leur besoin.
- Les services gérés, qui ont la charge de tout le processus de récolte de données. Les compagnies n'ont besoin que de fournir leurs besoins et le service est responsable de la récolte.

Ces solutions ont en commun le fait qu'elles reposent sur des configurations manuelles afin d'extraire les données dans une page Web. En effet, des opérateurs doivent détecter les attributs à extraire dans chacune des pages, et configurer le système en conséquence. De plus, les opérateurs doivent maintenir ces configurations, étant donné la nature évolutive des pages Web (Dilmegani, 2020). Dans la suite de ce mémoire, ces méthodes, reposant sur des configurations manuelles, seront référencées par le terme *solutions manuelles*.

1.4 Forums

Dans ce mémoire, l'attention est principalement portée sur les forums qui hébergent des communautés de cybercriminels. Ces plateformes sont des lieux privilégiés par les communautés d'acteurs illicites puisqu'elles offrent un environnement permettant d'échanger entre eux de façon anonyme (Macdonald *et al.*, 2015). Pour ce faire, un utilisateur doit se créer un profil sur un de ces forums. Par la suite, celui-ci peut ouvrir et commenter des sujets de discussions.

Les forums sont formés principalement de trois types de pages. Tout d'abord, on retrouve les pages contenant les profils des utilisateurs. Celles-ci contiennent des informations sur les divers

acteurs qui interagissent sur le forum. Chaque page est reliée à un seul enregistrement. Ensuite, on retrouve les pages contenant les différents sujets de discussions abordés sur le forum (figure 1.2). Ceux-ci sont divisés par catégorie et sont créés par les utilisateurs. Pour la suite, ce type de pages sera référé comme les pages de sujets de discussions. Chacune de ces pages contenant plusieurs sujets de discussions, et ceux-ci sont considérés comme étant des enregistrements (voir section 1.3). Les utilisateurs voulant interagir sur des sujets de discussions le font sur les pages de discussions. Chaque sujet de discussions est alors relié à une page de discussions. Les pages de sujets et de discussions sont de type multienregistrements.

Figure 1.2 Page représentant des sujets de discussions

Dans la grande majorité des cas, les sujets de discussions sont représentés sous forme de liste. Chacun des sujets de discussions contient minimalement le titre du sujet ainsi que son auteur et dans certains cas la date de publication. On retrouve aussi le nom du dernier utilisateur ayant commenté sur le sujet de discussions, la date de publication du dernier commentaire et certaines informations telles que le nombre de réponses au sujet. La figure 1.3 illustre trois captures d'écran de titre de sujets de différents forums.

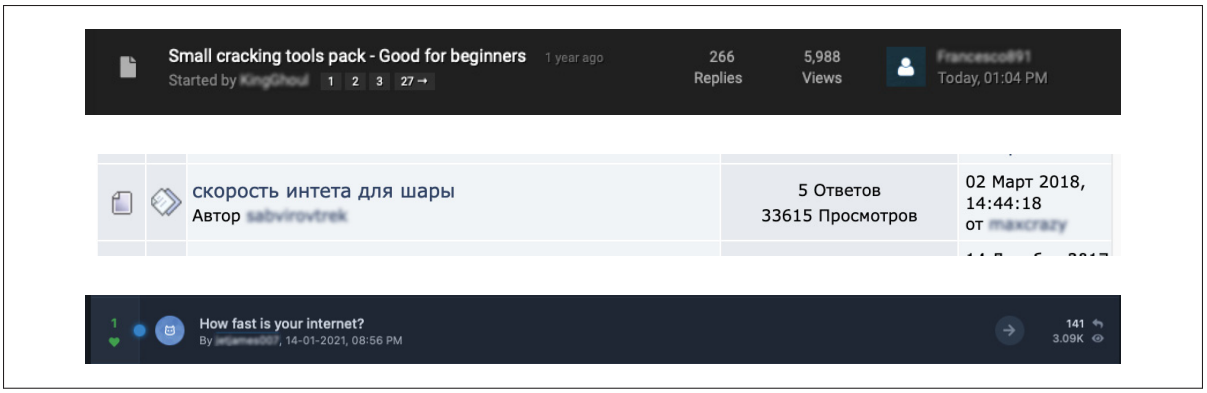


Figure 1.3 Exemple de sujets de discussions extrait de page Web contenant une série de titre de sujets de discussions

En plus des défis énumérés dans la section 1.3, l'extraction des sujets sur un forum apporte son lot de défis supplémentaires :

- Tout d'abord, le système doit être en mesure de détecter la frontière entre chaque sujet de discussion, que l'on peut aussi nommé enregistrement, afin d'associer les attributs correctement. En effet, l'objectif est d'associer le bon titre au bon auteur, et ainsi de suite. Ce défi se complique, car aucune norme n'existe sur le Web quant à la façon de séparer des enregistrements. La figure 1.4 représente les trois types d'organisation les plus répandus (division, liste, tableau). Cette liste n'est pas exhaustive, et chaque configuration est sujette à de multiples variations.
- Par leur nature, le contenu sur les forums est créé par des utilisateurs qui ont libre contrôle sur leur nom d'utilisateur ainsi que leur façon de s'exprimer. Cette liberté rend alors la tâche d'établir un vocabulaire fini difficile en raison des synonymes, fautes d'orthographe et autres éléments. Le tableau 1.1 illustre cette liberté de vocabulaire. De plus, la langue parlée sur un forum n'est pas unique. De ce fait, il peut arriver que des utilisateurs discutent dans des langues différentes de la langue principale du forum.

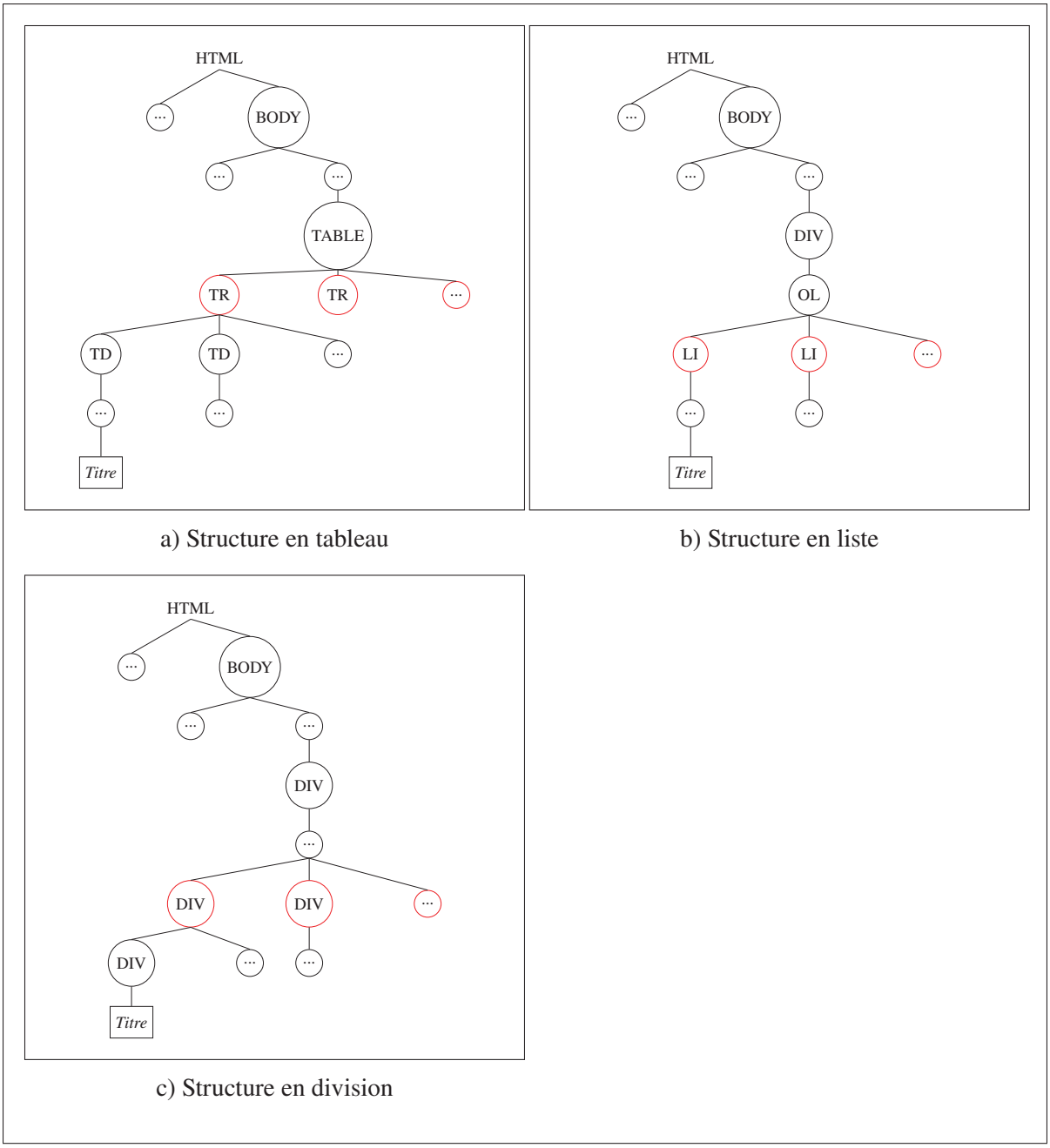


Figure 1.4 Les trois types de structures les plus souvent vues afin de séparer des enregistrements

- Un même enregistrement peut contenir des attributs semblables. De ce fait, le contexte autour de celui-ci est nécessaire afin de le classifier. Par exemple, la date de publication et la date du

Tableau 1.1 Exemple d'entrées utilisateur sur des forum

Titre	Acteur	Date
[☺□][MEGA] 700GB+ Of premium courses	Egau804	03-08-2021, 06 :14 PM
What do u think about Fortnite	matooou	1 year ago
universal SmartCard Sharing	t0KKe	Tuesday at 7 :06 PM

dernier commentaire ont le même format, tout comme le nom de l'auteur et celui du dernier ayant commenté le sujet de discussion.

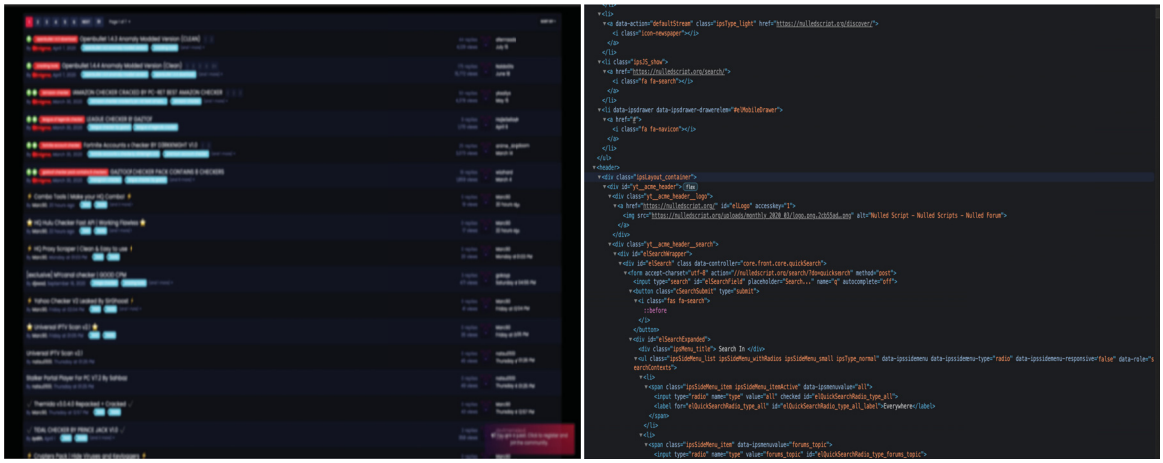
1.5 Questions et hypothèses

Typiquement, il est possible de représenter la structure d'une page Web de trois façons, tel qu'illustré à la figure 1.5 (Dallmeier, 2021). On retrouve la page rendue par le navigateur, la représentation HTML/CSS et le rendu sous forme d'arbre (voir section 2.1.1).

Puisque le contexte est important afin de déterminer à quel attribut appartient un extrait de texte dans une page Web, ce mémoire est dirigé par l'hypothèse qu'il est possible de traiter une page Web comme une séquence composée de balises HTML et de textes. De ce fait, l'identification des attributs dans une page peut se faire à l'aide de techniques empruntées au domaine de l'annotation de séquences (sequence labeling). Ces techniques permettent de prendre en compte le contexte d'une série de jetons (tokens) afin de catégoriser ceux-ci.

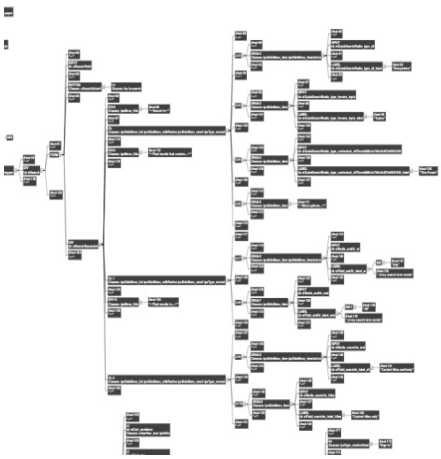
1.6 But de l'étude

L'objectif de ce mémoire est d'automatiser l'extraction de données sur un forum. Plus précisément, le but est de procéder à l'extraction des sujets de discussions sur une page et d'extraire le titre, l'auteur et date de publication pour chacun d'eux. De ce fait, deux sous-objectifs sont poursuivis, suivant l'hypothèse qu'il est possible d'utiliser des techniques d'annotation de séquences sur des pages Web.



a) Page Web telle que vue par l'utilisateur

b) Code source en HTML/CSS



c) Structure de la page sous forme d'arbre

Figure 1.5 Trois types de représentation d'une page Web

- Tout d'abord, identifier les attributs pertinents dans une page Web. L'identification des attributs devra tenir compte du contexte de ceux-ci afin de ne pas confondre l'auteur d'un sujet et le dernier utilisateur ayant commenté.
- Ensuite, extraire les attributs précédemment identifiés dans une page Web.

La méthode proposée doit à la fois fonctionner sur des forums présentement connus, mais aussi être en mesure d'être généralisée et ainsi être applicable sur de nouveaux forums qui n'ont pas été utilisés lors de l'élaboration de la solution. L'emphase est mise sur les forums qui accueillent des communautés de cybercriminels. De ce fait, la solution développée doit être multilingues, dont principalement en anglais et en russe.

La solution doit fonctionner sur des pages de type multienregistrements et doit aussi être adaptable pour fonctionner sur des pages de type un seul enregistrement. En effet, malgré que l'effort sera mis sur l'extraction des titres de sujets, il est important de prendre en compte que différents types de pages existent. En effet, la solution pourra éventuellement être adaptée pour les pages de discussions et de profils d'utilisateurs des forums, ainsi que celles d'autres plateformes comme les places de marchés.

D'un côté plus technique, la solution doit réduire l'intervention humaine requise afin d'ajouter une nouvelle source à surveiller ainsi que de la maintenir. De plus, le temps d'exécution au niveau de l'extraction doit être pris en compte afin d'être en mesure de surveiller en temps presque réel les communautés d'acteurs malveillants.

CHAPITRE 2

REVUE DE LITTÉRATURE

Ce chapitre vise à faire le lien entre la problématique de recherche et l'effort commun qui a été porté dans le domaine au cours des dernières années. De ce fait, la première partie de cette revue de littérature est consacrée au domaine du moissonnage du Web (section 2.1). Par la suite, le domaine de l'annotation de séquences est étudié (section 2.2), suivi des lectures en lien avec les différents sous-objectifs de cette recherche (section 2.3). Finalement, il est question des enjeux de généralisation de sa solution (section 2.4) ainsi que l'apport à la littérature de ce mémoire (section 2.5).

Il est important de noter que peu de travaux ont porté sur la problématique d'automatiser l'extraction des sujets d'un forum consacré aux communautés de cybercriminels. En effet, les applications dans le domaine du moissonnage du Web se sont surtout concentrées sur la comparaison de prix, le monitorage de la météo, le regroupement de contenus, la recherche et l'intégration (Singrodia *et al.*, 2019). De plus, le traitement d'une page Web comme une séquence unique est une nouvelle approche dans le domaine de l'extraction sur le Web.

2.1 Moissonnage du Web

Dans cette section, il est question des différentes techniques d'extraction automatique sur le Web qui ont été proposées par le passé. Selon Singrodia *et al.* (2019), ces techniques peuvent être divisées en trois catégories : moissonnage syntaxique (Syntatic Web Scraping), moissonnage sémantique (Semantic Web Scraping) et moissonnage par vision par ordinateur (Computer vision Web-page analyzing). Cette catégorisation a été reprise par Dallmeier (2021) dans ses travaux visant à proposer une méthodologie afin d'utiliser des outils de vision par ordinateur pour procéder à l'extraction de contenu sur des forums. Par conséquent, celle-ci a été adoptée dans ce mémoire.

Dans les prochaines sous-sections, chacune de ces techniques est présentée. De plus, pour chacune, des travaux de chercheurs ayant utilisé ces techniques à des fins d'extraction de contenu sur des forums sont présentés. Cette analyse permet de situer ce mémoire en comparaison avec les travaux d'autres chercheurs du domaine de l'extraction automatique sur le Web.

2.1.1 Moissonnage syntaxique

Le moissonnage syntaxique regroupe les techniques qui impliquent l'utilisation de la structure des pages Web à l'aide de patrons syntaxiques. Conséquemment, ces techniques tirent avantage du HTML Document Object Model (DOM), soit la représentation d'une page Web par un arbre racine étiqueté orienté (labeled ordered rooted tree) (Ferrara *et al.*, 2014). Plus précisément, chacune des balises HTML présentes dans le texte brut est représentée par un noeud afin de former une structure hiérarchique. Cette structure peut ensuite être exploitée afin de capturer des éléments spécifiques dans une page. La figure 2.1 représente la relation entre un fichier texte brut et le HTML DOM.

Parmi les techniques qui exploitent le HTML DOM afin de procéder à de l'extraction de contenu, on retrouve les Content Style Sheet Selectors (CSSSelector) ainsi que Xpath. Ces deux techniques permettent l'écriture de règles pointant vers un élément ou bien un ensemble d'éléments dans une page Web (Ferrara *et al.*, 2014). Par exemple, dans la figure 2.1a, il est possible d'exploiter le chemin $[html] \rightarrow [body] \rightarrow [h1]$ afin d'extraire le titre.

Les entreprises utilisant des solutions manuelles, telle l'entreprise Flare Systems, ainsi que des chercheurs, tels Nunes et MacDonald (Nunes *et al.*, 2016; Macdonald *et al.*, 2015) utilisent ces règles afin d'extraire du contenu à des fins d'extraction de contenu sur le Web. Cependant, ces règles offrent peu de flexibilité. En effet, elles sont fortement couplées à la structure actuelle de la page et peuvent être invalidées par le moindre changement apporté sur celle-ci. De plus, de nombreux sites Web ont des variations de structure interne (telle la présence, ou non, d'un attribut affiché dynamiquement), ce qui rend difficile l'écriture de règles qui prennent en compte

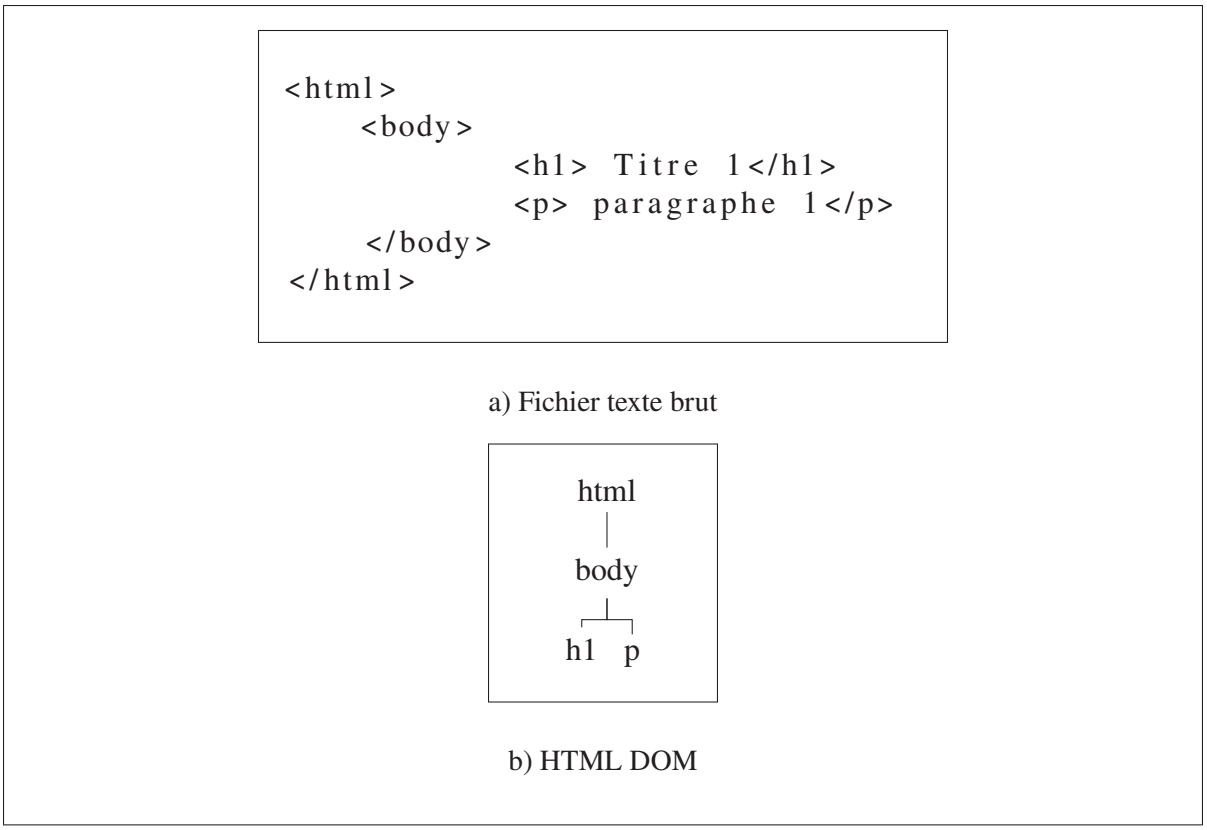


Figure 2.1 Représentation du HTML DOM à partir du texte brut

ces variations. D'autre part, ces variations peuvent parfois être difficiles à détecter au premier coup d'oeil, ce qui complique la tâche des opérateurs.

Suite à ces constatations, des chercheurs ont exploité des algorithmes de correspondance d'arbres (tree matching algorithms) afin de proposer des méthodes plus robustes aux changements (Ferrara *et al.*, 2014). Ces techniques demandent cependant une grande puissance de calcul (Cording, 2011a), ce qui en fait de mauvais candidats afin de procéder à la surveillance en temps réel de pages Web.

2.1.2 Moissonnage sémantique

Ces méthodes impliquent l'utilisation de contenu, soit le texte contenu dans les pages Web, afin de généraliser des règles d'extraction. De ce fait, des techniques de forage de textes sont

utilisées. C'est le cas entre autres du logiciel W4F qui assiste un opérateur en générant des expressions régulières à partir des régions identifiées (Ferrara *et al.*, 2014). Cette méthode est cependant peu utilisée due au manque de flexibilité des expressions régulières. En effet, Ferrara *et al.* (2014) mentionnent que chaque expression régulière risque de cesser de fonctionner suite à un changement mineur dans le contenu de la page Web.

Les caractéristiques textuelles, telles que la longueur du texte, la présence de certains mots-clés et le vocabulaire utilisée, peuvent être utilisées pour entraîner un modèle de classification (Dallmeier, 2021). Cette classification est principalement utilisée afin d'extraire des régions clés dans une page, comme le corps d'un article (Zhou & Mashuq, 2014). De plus, il est possible de combiner ces caractéristiques avec des informations provenant du HTML DOM afin d'entraîner des classificateurs plus avancés (Carle, 2020). Ces méthodes sont cependant difficiles à généraliser sur des sites Web non utilisés durant l'entraînement.

Afin d'extraire les sujets d'un forum, Baskaran & Ramanujam (2018) proposent une méthode qui permet d'extraire des sujets d'utilisateurs sur un forum médical à l'aide de règles sémantiques. Ces règles sont définies manuellement et reposent grandement sur une morphologie standardisée. Par exemple, les titres doivent contenir un pourcentage de mots dont la première lettre est une majuscule. Zhang, Jin, Lin & Gong (2012), quant à eux, proposent la détection des termes fréquents afin de différencier le contenu généré par des utilisateurs du contenu statique. Cette méthode est efficace afin de distinguer les différents enregistrements, mais ne permet cependant pas de distinguer les différents attributs lors de l'extraction.

2.1.3 Moissonnage par Vision par Ordinateur

Cette catégorie regroupe les techniques utilisant les éléments visuels d'une page Web à des fins de classification et d'extraction. Ces éléments peuvent provenir directement du HTML DOM qui, une fois affiché dans le navigateur, contient les informations sur la position et la grandeur de chaque section de la page (tel le menu de navigation, le contenu principal, etc.,), aussi communément appelé bloc. Comme il faut interpréter chaque page Web à l'aide d'un

navigateur pour avoir accès à ces informations, ces techniques demandent plus de puissance de calcul.

L'utilisation de ces caractéristiques a permis à Liu, Yan & Xiao (2011) de développer une méthode afin d'identifier les blocs contenant des commentaires d'utilisateurs dans des forums en utilisant des algorithmes de similarité. Tout comme la technique de moissonnage sémantique de Zhang *et al.* (2012), cette méthode demande un nombre minimal d'entrées sur une même page afin de fonctionner et ne distingue pas les différents attributs de chaque publication.

2.2 Annotation de séquences

Selon l'hypothèse de recherche, il est possible d'identifier et d'extraire des attributs dans une page Web à l'aide de techniques d'annotation de séquences. De ce fait, cette section est une introduction aux techniques d'annotation de séquences en traitement automatique des langues naturelles. Par conséquent, l'utilisation de ces techniques dans le domaine du Web est discuté dans la section 2.3, avant de conclure avec l'introduction des champ aléatoires conditionnels (conditional random fields ou CRF) dans la section 2.2.3.

2.2.1 Annotation de séquences appliquée au domaine du Web

L'utilisation de techniques d'annotation de séquences à des fins d'extraction sur une page Web est un nouveau domaine. En effet, ces techniques sont principalement utilisées dans des tâches de reconnaissance d'entités, d'étiquetage morpho-syntaxique (part-of-speech tagging) ainsi que de segmentation de texte (text chunking) (He *et al.*, 2020). Ces méthodes reposent en grande partie sur un ensemble de textes où chaque jetons (tokens) est annoté selon sa catégorie.

Dans leurs travaux d'identification de produits sur des forums, Portnoff *et al.* (2017a) notent que les outils d'annotation automatique disponibles sur le marché sont spécialisés pour des domaines spécifiques (articles de journaux, encyclopédie, etc.). De plus, ils constatent que ces outils sont surtout adaptés aux textes standards tels que ceux retrouvés sur Wikipédia et dans les journaux,

en opposition aux textes bruités retrouvés dans les médias sociaux. Par conséquent, les solutions actuelles ne sont pas adaptées aux vocabulaires présents sur les forums.

La problématique des mots mal orthographiés et inconnus a été abordée par Nunes *et al.* (2016). Lors de leurs expériences de classification d'annonces provenant de forums, l'utilisation des n-grammes de caractères (character n-grams) a surpassé une approche plus traditionnelle de racinisation (word stemming). Kadoguchi *et al.* (2019), quant à eux, ont procédé à plusieurs prétraitements du texte, tels que le nettoyage (retrait des nombres et parenthèses), la normalisation des mots (word normalization), la racinisation et le retrait des mots vides (stop words removal).

Finalement, des techniques d'annotation de séquences ont été appliquées sur des pages Web, mais à des fins différentes que l'extraction d'information. Par exemple, Macdonald *et al.* (2015) combine l'utilisation d'un système d'étiquetage morpho-syntaxique à des outils d'analyse de sentiments afin de détecter les publications de forum ciblant des infrastructures à risque (Macdonald *et al.*, 2015). Plus globalement, Wicaksono & Myaeng (2013) utilisent ces méthodes afin de procéder à l'identification de conseils sur des forums en ligne. Ces deux approches reposent sur des données préalablement extraites de forums et annotées manuellement.

2.2.2 Mise en jetons par codage de paire d'octets (Byte-Pair Encoding) (BPE)

Afin de procéder à l'annotation de séquences, l'une des étapes est de diviser le texte en sous-unités, appelés jetons (tokens). Ce sont ces jetons qui seront par la suite annotés. Une des façons de procéder est d'utiliser les espaces afin de séparer et de délimiter chaque jeton. Cependant, cette technique n'est pas adaptée pour les mots rares, ou bien les mots qui ne sont pas présents dans l'ensemble d'entraînement. En effet, le modèle n'aura pas assez d'exemples d'utilisation de ces mots (voire aucun dans le cas des mots non présents dans l'ensemble d'entraînement) et ne pourra donc pas les apprendre (Heinzerling & Strube, 2017). Cette caractéristique des mots rares ou inconnus est particulièrement présente dans le cas où le texte à annoter provient de forums de discussion sur le Web (voir section 1.4). Par exemple, les noms d'utilisateurs ainsi que le vocabulaire utilisé par les acteurs de ces forums.

Afin de régler le problème des mots-rares ou inconnus, Sennrich, Haddow & Birch (2016) ont proposé un algorithme basé sur l'algorithme «byte pair encoding» afin de créer un vocabulaire composé de sous-mots (subwords). Ces sous-mots permettent de représenter les mots rares et inconnus, en séparant ceux-ci en plus petites séquences. Il est alors possible d'entraîner un modèle permettant la mise en jetons (tokenizateur) de type BPE avec un corpus provenant d'un domaine précis.

2.2.3 Champ aléatoire conditionnel

Un des défis lors de l'extraction automatique sur le Web est l'importance du contexte lors de l'identification des attributs (voir section 1.4). C'est d'ailleurs pour cette raison que l'hypothèse de ce mémoire cible l'utilisation des techniques d'annotation de séquence afin de procéder à l'identification des attributs dans une page Web. En effet, l'une des approches d'annotation de séquence consiste à utiliser un modèle discriminant, tels les champs aléatoires conditionnels.

Tel que mentionné dans Wallach (2004), les champs aléatoires conditionnels permettent d'étiqueter une séquence $\boldsymbol{x}$ en sélectionnant la suite d'étiquettes $\boldsymbol{y}$ qui maximise la probabilité conditionnelle $p(\boldsymbol{y}|\boldsymbol{x})$. En d'autres mots, ceux-ci permettent de prendre en considération le contexte de chaque jeton composant une séquence afin de procéder à l'annotation de celles-ci. Cette caractéristique différencie les champs aléatoires conditionnels des modèles génératifs (telles les chaînes de Markov), qui assument que chaque observation est indépendante (Quattoni, Wang, Morency, Collins & Darrell, 2007).

Par conséquent, il est possible d'utiliser les champs aléatoires conditionnels pour prédire la classe de chaque jeton exprimée avec une notation «BIO» (soit **B** pour début , **I** pour intérieur (Inside) et **O** pour extérieur (Out)). Dans la figure 2.2, chaque jeton composant la phrase «La Tour Eiffel est à Paris» a été étiqueté avec cette annotation. Les jetons faisant partie de la sous-séquence «Tour Eiffel» ont été identifiés comme étant un lieu, le jeton «Paris» comme étant une ville, et le reste des jetons comme appartenant à aucune catégorie.

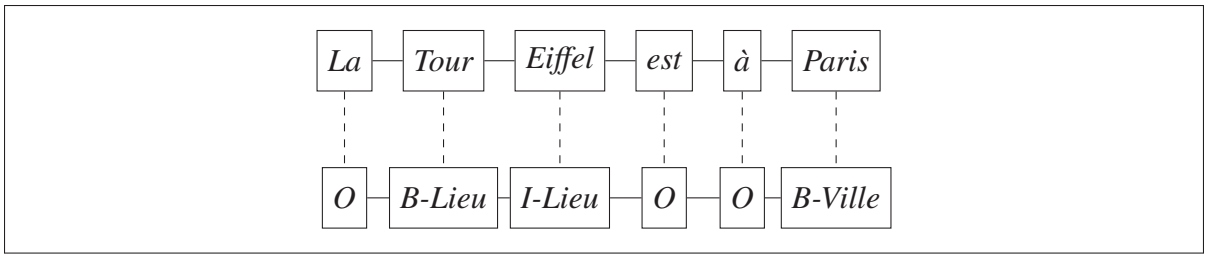


Figure 2.2 Exemple d'une séquence avec annotation BIO

2.3 Identification et extraction d'attributs par annotation de séquences

Dans la section précédente, le domaine de l'annotation de séquences a été introduit, en plus des champs aléatoires conditionnels. Dans celle-ci, il est question de faire le pont entre ces techniques et les sous-objectifs de ce mémoire. La section 2.3.1 s'attarde aux recherches dont les travaux ont porté sur l'identification d'attributs dans une page Web à partir de son texte. Par la suite, il est question, dans la section 2.3.2, de l'étude d'algorithmes permettant l'extraction d'enregistrements et ses attributs d'une page Web.

2.3.1 Identification des attributs dans une page Web

Le premier sous-objectif de ce mémoire est l'identification des attributs présents dans une page Web en utilisant des techniques d'annotation de séquences. Tel que mentionné à la section 2.2.1, aucun travail connu n'utilise ces techniques afin de procéder à de l'extraction de contenu sur le Web. Cependant, il est intéressant d'étudier certains travaux qui utilisent le texte présent dans les pages HTML afin de procéder à l'identification des attributs.

C'est le cas de Ujwal, Gaind, Kundu, Holla & Rungta (2017), qui utilisent l'algorithme de classification Random Forest. L'algorithme développé prend en entrée le texte contenu dans chacun des noeuds du HTML DOM, et une prédiction de la classe de sortie est alors produite pour chacun d'eux (Ujwal *et al.*, 2017). La méthode proposée dans ce mémoire est similaire à cette dernière, à la différence que les pages Web ne seront pas traitées comme un ensemble de

noeuds contenant du texte, mais bien comme une série de jetons. Chaque jeton étant du texte provenant de la page Web ainsi que les balises HTML.

La problématique du bruit autour des attributs n'a pas été abordée dans la littérature. En effet, l'hypothèse généralement acceptée stipule qu'un attribut est délimité par des balises HTML. Cependant, dans des situations réelles, ce n'est pas toujours le cas tel qu'illustré dans la figure 2.3 où le nom de l'acteur (Acteur_X) est précédé du le préfixe "Par". Afin d'éliminer ce type de bruit, l'entreprise Flare Systems utilise des expressions régulières. Toujours dans la figure 2.3, l'expression régulière *r"Par \s(\w+)"* permet de retirer le mot *Par* avant le nom de l'acteur. Cette technique est efficace, mais requiert une intervention manuelle. De plus, comme les règles Xpath et CSSSelector, les expressions régulières peuvent être invalidées lors d'un changement de format au niveau du site Web.

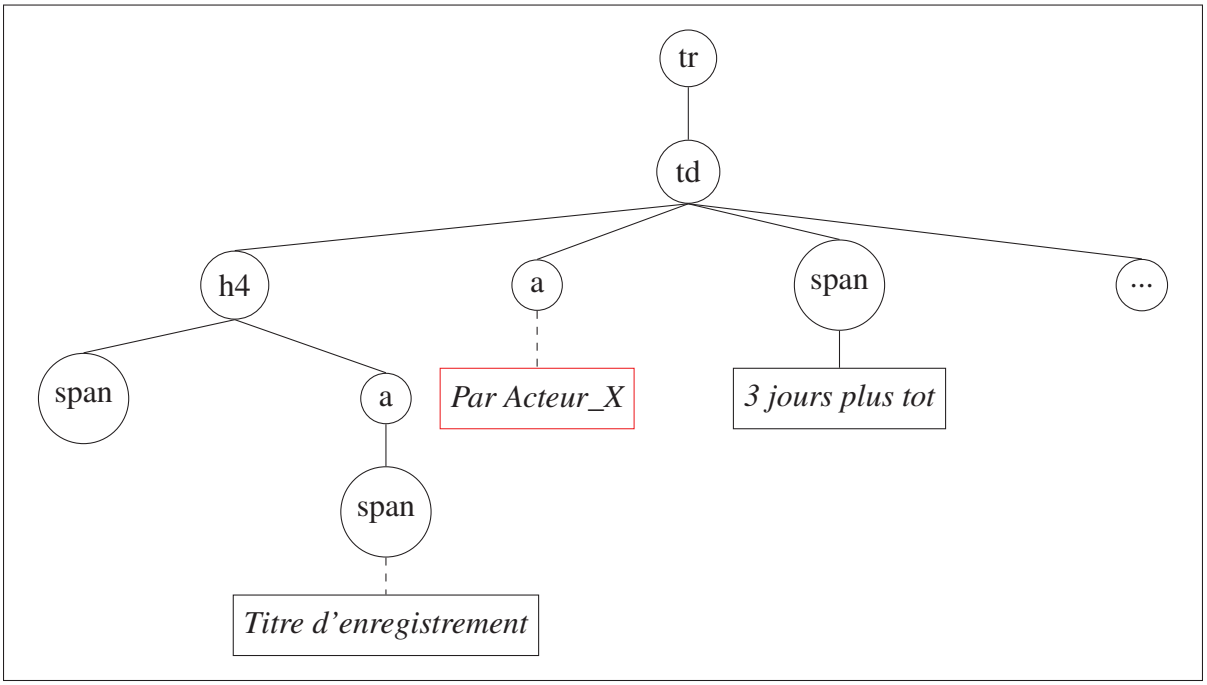


Figure 2.3 Enregistrement dont le nom de l'auteur contient du bruit

Finalement, de nombreuses informations présentes dans les pages Web ne sont pas pertinentes à des fins d'identification de contenus. C'est le constat qu'a fait Cording (2011a) lors de ses travaux consistant à tester l'utilisation des algorithmes d'alignement d'arbres dans ce contexte

(Cording, 2011a). Celui-ci élimine les branches du HTML DOM qui n'ont pas de chance de se retrouver dans le patron recherché. De son côté, Ujwal *et al.* (2017) utilisent des expressions régulières afin de normaliser le document en remplaçant les dates présentes dans le texte par un mot clé «$DATE». De plus, il retire les termes les plus fréquents, selon l'hypothèse que ceux-ci font partie du gabarit de la page Web (et ne font alors pas partie des éléments à extraire). Puisque le contexte est important afin de classifier certains attributs (voir section 1.4), cette technique n'est pas compatible avec l'extraction de sujets de discussions.

2.3.2 Extraction des enregistrements et des attributs dans une page Web

Le deuxième sous-objectif de ce mémoire consiste à extraire les attributs préalablement identifiés dans une page Web. Le principal défi ici est de rassembler les attributs d'un même enregistrement (dans le cas d'une page multienregistrements). De ce fait, il est intéressant d'étudier des travaux qui touchent aux problèmes d'identification des frontières entre chaque enregistrement.

À la section 2.1, certaines approches visant l'extraction des enregistrements ont été mentionnées. Celles-ci reposent cependant sur des techniques de vision par ordinateur, des règles de sémantique spécifiques à un domaine ou bien des algorithmes de similarités. Ces techniques demandent donc du temps de traitement élevé ou bien une intervention humaine dans le cas des règles de sémantique. Comme la solution doit être en mesure d'extraire rapidement les informations ciblées tout en minimisant l'effort humain, ces approches ne sont pas retenues pour la solution finale.

De ce fait, il est pertinent d'étudier les approches se servant de la structure du HTML DOM ainsi que du contenu de la page. De plus, l'hypothèse qu'un groupe d'enregistrements partage un noeud parent est fortement accepté dans le domaine. Par conséquent, Zhang *et al.* (2012); Ujwal *et al.* (2017) utilisent les attributs trouvés préalablement dans une page Web afin de déterminer le noeud parent de tous les enregistrements. La figure 2.4 illustre ce principe. Cette approche demande cependant un minimum d'enregistrements afin de déterminer les noeuds parents. De

ce fait, Zhang *et al.* (2012) retirent les pages ayant moins de 10 enregistrements de son jeu d'entraînement.

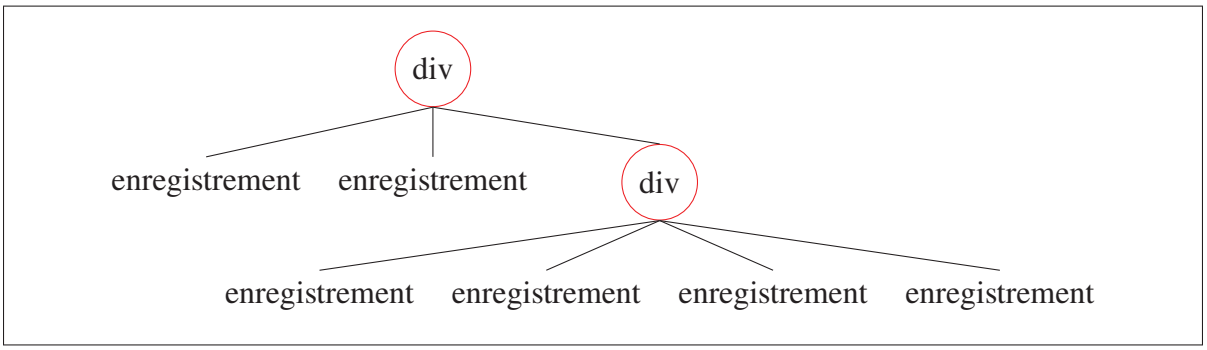


Figure 2.4 Deux noeuds représentant des parents d'enregistrements

Finalement, Ujwal *et al.* (2017) introduisent le concept d'attributs optionnels et obligatoires. Par conséquent, un enregistrement doit contenir minimalement tous les attributs obligatoires afin d'être valide. De plus, Ujwal *et al.* (2017) introduisent un algorithme permettant de compenser pour les erreurs de classification. En effet, il se peut que lors du processus d'identification des attributs, qu'un même enregistrement contienne plus d'une occurrence d'un même attribut suite à une erreur. Lors du processus d'extraction, afin de déterminer quelle occurrence est la plus susceptible d'être la bonne, Ujwal *et al.* (2017) compare les chemins d'accès de chaque occurrence avec les chemins d'accès des mêmes occurrences au sein des autres enregistrements. Le chemin le plus fréquent est alors déterminé gagnant.

2.4 Généralisation de la solution

Un des objectifs de ce mémoire consiste à développer une solution en mesure de fonctionner à la fois avec les forums utilisés lors de l'entraînement, mais aussi avec des forums jusqu'alors inconnus. En apprentissage machine, ce concept se nomme la généralisation. Appliqué dans le domaine de l'extraction automatique sur le Web, celui-ci peut se diviser en deux grandes catégories : source unique et multisource.

Tout d'abord, l'approche source unique permet d'entraîner un modèle de classification spécifique à une source (site Web). De ce fait, deux approches en découlent. La première est de recueillir les pages Web d'une source à un instant donné et de diviser celles-ci en un jeu d'entraînement, de validation et de test (Zhou & Mashuq, 2014). Pour la deuxième méthode, il faut identifier une source ayant subi des changements de structure au cours du temps. Le modèle est alors entraîné sur une version et évalué sur une version future de la même source (Carle, 2020; Ujwal *et al.*, 2017).

L'approche multisource, tout comme l'approche à source unique, se divise en deux catégories. Tout d'abord, la première approche consiste à entraîner le modèle sur une fraction de toutes les pages recueillies (peu importe la source) pour ensuite tester celui-ci sur le reste (Zhou & Mashuq, 2014). Pour la suite de ce mémoire, cette approche sera référencée par l'approche multisource fractionnaire. La deuxième approche consiste à entraîner la solution sur un sous-ensemble des sources et à tester sur les sources restantes (Zhou & Mashuq, 2014; Manica, Dorneles & Galante, 2017). Par la suite, celle-ci sera référencé par approche multisource classique.

Dans ses travaux, Zhou & Mashuq (2014) comparent trois approches afin de tester leur solution, soit l'approche source unique intemporelle ainsi que les deux approches multisources (fractionnaire et classique). Ceux-ci ont testé leurs solutions sur un ensemble composé de 349 pages Web provenant de divers sites Web de nouvelles (The Verge, USA Today, etc.). Ils obtiennent 100% pour la mesure combinée F1 (voir section 3.2.3) avec l'approche à source unique, mais tombent à 92,83% pour l'approche multisource fractionnaire et à 31,28% pour l'approche multisource classique. Zhou & Mashuq (2014) conclurent alors que leur modèle est en mesure d'apprendre la structure de plus d'une source à la fois, mais ne peut généraliser sur des sources inconnues.

2.5 Apport à la littérature

La recherche présentée dans ce mémoire se distingue des autres recherches ayant comme sujets principaux l'extraction automatique sur le Web. En effet, il a été montré, au travers de cette revue

de littérature, que l'utilisation des techniques d'annotation de séquences à des fins d'extraction automatique sur le Web constitue un nouveau domaine de recherche. Conséquemment, les articles étudiés faisaient état des derniers développements dans le domaine tout en faisant des liens avec les objectifs initiaux.

Les constatations qui en découlent sont nombreuses :

- Tout d'abord, les solutions qui permettent l'extraction des attributs individuels sont, pour la plupart, spécifiques à un site Web d'une version donnée (section 2.1).
- Ensuite, aucune des solutions connues ne permet de transformer une page Web en une séquence continue de jetons, permettant l'utilisation d'outils d'annotation de séquences. De plus, les outils disponibles afin de procéder à l'annotation de séquences ne sont pas adaptés au vocabulaire présent sur les forums (section 2.2.1).
- Finalement, les outils étudiés reposant sur une classification automatisée ne tiennent pas compte du contexte. De ce fait, ces approches se contentent d'extraire des attributs pour lesquels il ne peut y avoir d'ambiguïté, comme le corps d'un article section (2.3.1).

Au final, aucune solution étudiée ne permet d'extraire directement l'auteur, le titre et la date de publication d'un sujet à l'aide d'outils d'intelligence artificielle. La solution qui se rapproche le plus est celle de Baskaran & Ramanujam (2018), mais celle-ci est basée sur des règles de sémantique associées au domaine médical. De plus, il n'est pas spécifié comment la distinction entre les attributs similaires, tel que le nom de l'auteur et le nom de la dernière personne ayant commenté est effectué.

CHAPITRE 3

MÉTHODOLOGIE

La méthodologie présentée dans ce chapitre est divisée en trois parties. La première partie présente l'acquisition et le traitement des données (section 3.1). Par la suite, il est question des étapes menant à la résolution du premier sous-objectif, soit l'identification des attributs dans une page Web à partir de techniques d'annotation de séquences (section 3.2). Finalement, l'extraction des enregistrements et de ses attributs, en lien avec le deuxième sous-objectif, est abordé dans la section 3.3. La figure 3.1 illustre la division de la méthodologie. Celle-ci est de type méthode expérimentale.

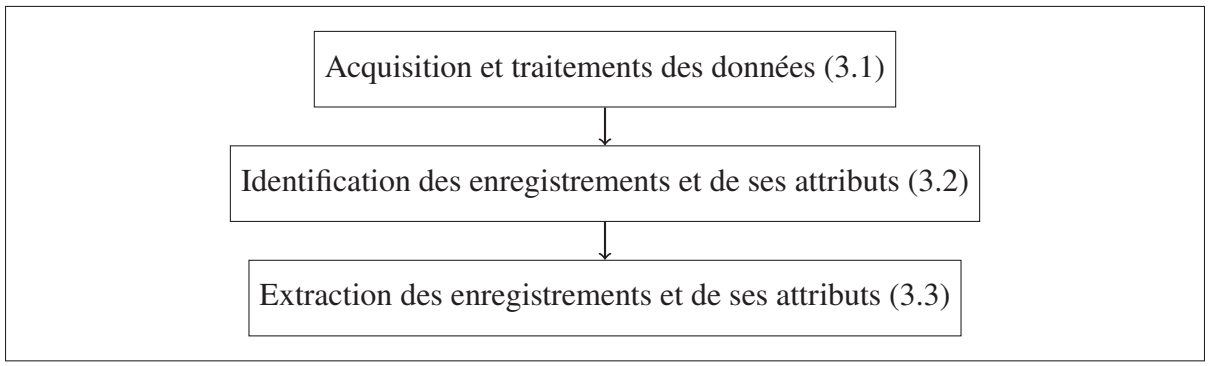


Figure 3.1 Structure de la méthodologie

3.1 Acquisition et traitements des données

Dans la première section de ce chapitre, il est question de l'acquisition des données et du traitement de celles-ci. La définition des données disponibles sera effectuée dans la section 3.1.1. Par la suite, l'annotation de celles-ci en vue de procéder à l'entraînement d'un modèle d'annotation de séquences est discuté (section 3.1.2). Finalement, la distribution (3.1.3), la validation (3.1.4), la structure et le nettoyage (section 3.1.6) des données sont abordées (section 3.1.5).

3.1.1 Définition des données

Idéalement, afin de procéder à la réalisation des objectifs de ce mémoire, un ensemble de données annotées contenant tous les forums hébergeant des cybercriminels serait utilisé. Cependant, ce type de base de données n'est pas disponible. Par conséquent, les données utilisées proviennent de l'entreprise Flare System sous forme de "Web ARChive" (WARC). Chaque jour, une archive WARC est créée pour chacune des sources (sites Web) surveillées par l'entreprise. Chaque archive contient alors les pages HTML qui ont été récoltées lors d'une journée de travail.

Le tableau 3.1 illustre les sites Web de types forum présents dans les archives de l'entreprise. Parmi ces archives, on retrouve 29 sources de type forum. Il est à noter que le nombre d'exemples (pages Web) disponibles pour chaque source n'est pas uniforme. De plus, la langue principale de chacune varie, ainsi que la présence de l'attribut date de publication et la structure des pages contenant les sujets de discussions (division, liste et tableau). En moyenne, une page Web contient environ 20 sujets de discussions, mais ce nombre peut varier grandement en fonction du site Web.

Pour chaque archive, on retrouve le nombre de pages Web contenant des sujets de discussions, la langue principale, la structure (section 1.4) ainsi que la présence de l'attribut date de publication. De plus, il est important de noter que le nombre d'exemples diffère grandement entre les sources. Ceci s'explique tout d'abord par la popularité du forum, mais aussi par des enjeux techniques du domaine de l'exploration du Web (disponibilité du site Web, systèmes anti-robots) (voir section 1.3).

Afin qu'une source ne soit pas surreprésentée dans l'ensemble de données, un maximum de 1 200 exemples de chaque source est utilisé. En effet, l'objectif est de proposer une méthode qui est en mesure de généraliser l'extraction de contenu sur de nouveaux sites Web. Par conséquent, si une source est présente en trop grand nombre, le modèle risque de se coller sur la structure de cette source (surapprentissage).

Tableau 3.1 Échantillon des sources contenus dans les archives de l'entreprise Flare Systems

Source	Pages Web utilisées	Langue	Structure	Attribut «Date de publication»
S1	596	Anglais	Division	X
S2	1 200	Russe	Division	X
S3	1 200	Russe	Division	X
S4	1 079	Russe	Division	X
S5	739	Anglais, Russe	Tableau	
S6	1 200	Anglais	Tableau	X
S7	1 200	Anglais	Tableau	
S8	1 200	Anglais, Multiples	Liste	X
S9	1 200	Anglais	Tableau	
S10	1 200	Anglais	Division	X
S11	895	Anglais	Division	X
S12	1 200	Anglais	Division	X
S13	1 200	Russe	Division	X
S14	484	Anglais, Russe	Liste	X
S15	68	Français	Divison	
S16	279	Anglais	Tableau	
S17	1 200	Anglais, Russe	Tableau	
S18	130	Anglais	Tableau	
S19	1 200	Anglais	Division	X
S20	1 200	Russe	Division	X
S21	86	Anglais	Division	X
S22	253	Russe	Tableau	
S23	1 200	Anglais	Liste	X
S24	1 200	Anglais	Division	X
S25	1 200	Anglais	Tableau	X
S26	290	Anglais	Division	X
S27	948	Anglais	Liste	X
S28	198	Français , Anglais	Liste	
S29	1 200	Anglais	Tableau	X

En plus de ses archives, l'entreprise donne accès aux règles d'extraction qu'elle utilise afin d'extraire les différents attributs pour chacune des sources. Ces règles sont de types CSSSSelector et XPath. De plus, celles-ci sont combinées à des expressions régulières afin d'extraire la partie

pertinente des attributs présents dans les pages Web (voir section 2.3.1). Ces règles sont utilisés afin d'annoter l'ensemble de données (section 3.1.2).

Le tableau 3.2 illustre les règles permettant l'extraction de chacun des attributs dans un titre de sujet d'une source. L'attribut enregistrement est la règle permettant de délimiter les différents sujets de discussions entre eux.

Tableau 3.2 Exemples de règles d'extraction de sujets de discussions d'une source

attribut	Règle
Enregistrement	XPathExtractor("//table[@id='threadslist']/tbody[contains(@id,'threadbits')]/tr[count(./td)>1]")
Titre	XPathExtractor(".//td[contains(@id,'threadtitre')]//a")
Auteur	XPathExtractor(".//td[contains(@id,'threadtitre')]/span[contains(@onclick, 'member')]")
Date de publication	CSSExtractor("div.thread-info", regex="By.*,\s(.*,\s.*)")

Finalement, un corpus composé des pages Web provenant d'une archive WARC comprenant une journée de travail est utilisé afin de définir un vocabulaire adapté aux sites Web peuplés par des communautés de cybercriminels (voir section 2.2.2). Ce corpus est par la suite utilisé pour entraîner un outil de segmentation avec la méthode BPE (byte pair encoding) et sera ainsi nommé corpus BPE.

3.1.2 Annotation des données

Afin de pouvoir procéder à l'entraînement d'un modèle d'annotation de séquences, les données doivent être annotées. Pour ce faire, les règles XpathSelector et CSSSelector sont utilisées afin d'identifier les attributs. De plus, les expressions régulières sont utilisées afin d'identifier la partie de la séquence pointée par la règle qui fait partie de l'attribut. Si aucune règle n'est donnée, toute la séquence est considérée comme étant l'attribut. Ce comportement est illustré à la figure 3.5.

Les attributs identifiés sont :

- Titre
- Auteur

- Date de publication
- Enregistrement (soit le noeud parent de l'enregistrement)

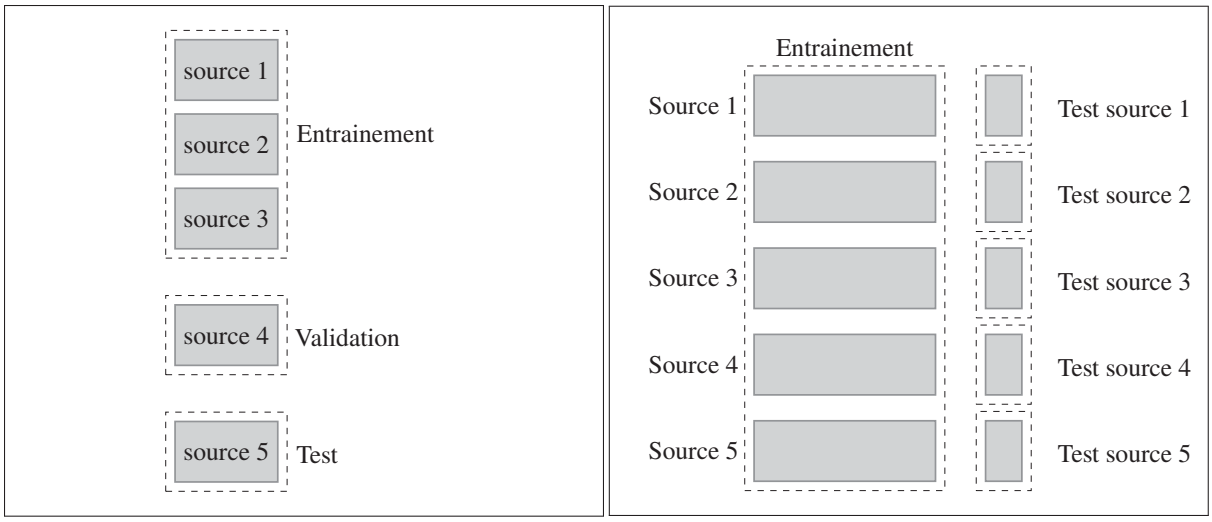
3.1.3 Distribution des données

Afin de procéder à l'entraînement d'un modèle d'annotation de séquences, l'ensemble de données doit être divisé afin d'obtenir un jeu d'entraînement, un jeu de validation et un jeu de test. Le jeu de validation est utilisé afin d'ajuster les hyperparamètres tandis que le jeu de test permettra de mesurer les performances de la solution.

Comme vues à la section 2.4, il y a deux grandes façons de procéder afin de créer le jeu d'entraînement, soient les approches à sources uniques et multisource. Dans ce mémoire, comme l'objectif est de développer une solution pouvant être généralisée sur des sources non vues durant l'entraînement, l'approche multisource classique est privilégiée, car elle permet de mesurer les performances de la solution sur des sources non vues durant l'entraînement. Par conséquent, les sources seront divisées selon quatre critères afin de créer l'ensemble A (figure 3.2a) :

- La langue principale du forum
- Le nombre de pages valides disponible
- La présence ou non de l'attribut "date de publication"
- La structure de la page Web (Division, Liste, Tableau)

Suite à l'expérience multisource classique, le modèle retenu est testé lors d'une expérimentation multisource fractionnaire. Pour ce faire, chacune des sources est divisée de façon aléatoire en un jeu de données d'entraînement et de test (80%, 20%) afin de créer l'ensemble B (figure 3.2b). Le modèle est entraîné sur un jeu de données d'entraînement contenant 80% de chacune des sources. L'évaluation est effectuée individuellement pour chacune des sources, avec le 20% de page Web n'ayant pas servi lors de l'entrainement.



a) Ensemble A b) Ensemble B

Figure 3.2 Exemple de la division des ensembles A et B avec un échantillon initial de 5 sources

3.1.4 Validation des données

Tel que mentionné à la section 2.1.1, l'utilisation des règles CSSSelector et Xpath pour d'identifier les attributs n'est pas infaillible. Par conséquent, afin de s'assurer que chaque échantillon soit bien annoté, un algorithme vérifie pour chacun que :

- Il y a présence d'enregistrements
- Chacun des attributs ayant une règle d'extraction sont présents
- Il y a le même nombre de chaque attribut (Cela vient du fait que chaque enregistrement contient une fois chaque attribut).

Ce mécanisme ne peut cependant pas détecter si des enregistrements sont présents dans une page Web, mais n'ont pas été identifiés. Par conséquent, pour chacune des sources, un échantillon aléatoire est vérifié manuellement, soit environ une dizaine de pages Web par source choisie au hasard.

3.1.5 Structuration des données

Afin de faciliter l'utilisation de techniques d'annotation de séquences, les pages Web sont tout d'abord transformées, passant d'un état brut (texte sous forme de code HTML) en un format structuré. Ainsi, pour chaque page HTML, une structure en arbre est créée. Pour la suite, cette structure est référée par *graphe_HTML* (représentée à la figure 3.3). Chaque noeud représente une balise HTML et les feuilles contiennent le texte.

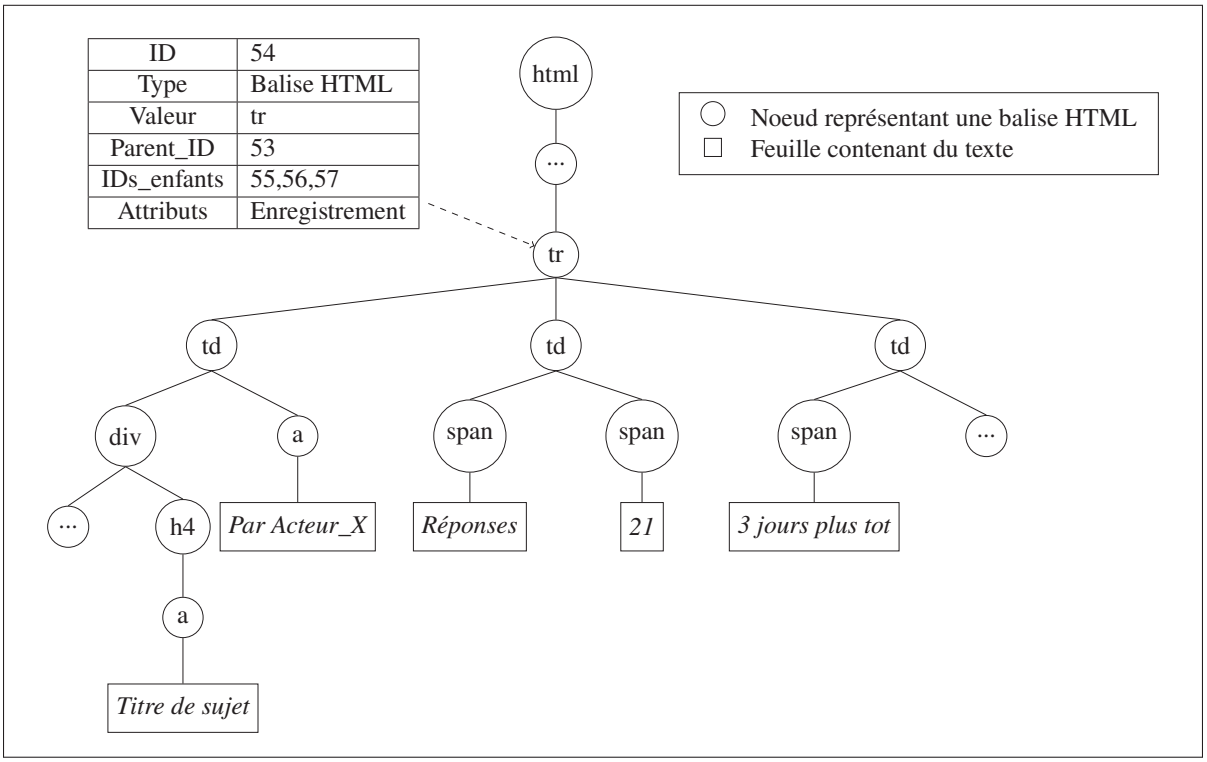


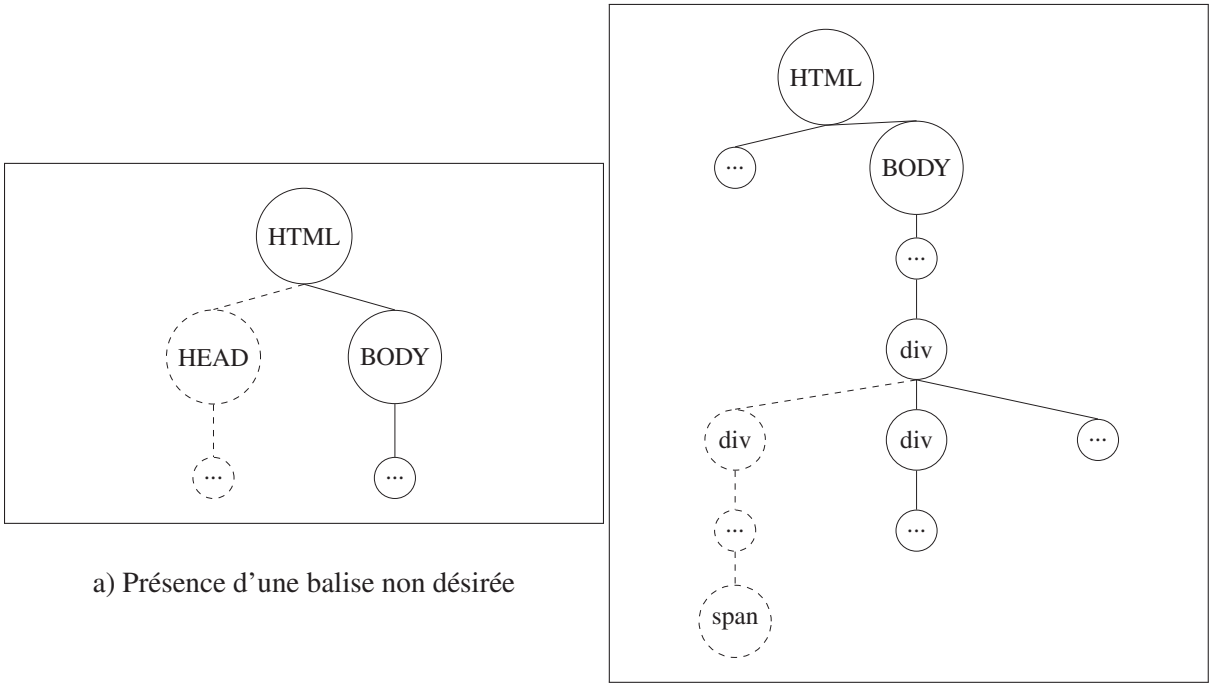
Figure 3.3 Extrait d'un *graphe_HTML*

3.1.6 Nettoyage des données

Dans le but de réduire le bruit et la taille des *graphe_HTML*, ceux-ci sont nettoyés en coupant des branches. Ce procédé est nécessaire afin de réduire la taille des séquences, et ainsi, aider le modèle à se concentrer sur les éléments pertinents des pages Web.

De ce fait, une branche est coupée lorsqu'un de ses enfants répond à l'une des conditions suivantes :

- Condition 1 : Présence d'une balise HTML *head*, *style* ou *meta* (figure 3.4a).
- Condition 2 : La feuille de la branche est une balise HTML (figure 3.4b).



a) Présence d'une balise non désirée

b) Présence d'une feuille qui n'est pas du texte

Figure 3.4 Des *graphe_HTML* dont les branches en pointillé ont été coupées

Ces règles permettent de réduire considérablement la taille de certaines pages tout en évitant de retirer des informations pertinentes à l'identification des attributs. En effet, les informations contenues dans les balises *head*, *style* et *meta* ne sont pas liées au contenu présent sur la page. En général, les balises *head* et *meta* contiennent des métadonnées tandis que la balise *style* renferme des informations dédiées au moteur de rendu. Finalement, les branches qui ne contiennent pas de texte n'ajoutent rien au contexte et peuvent alors être retirées. Celles-ci vont souvent contenir des images ou bien des publicités.

3.2 Identification des enregistrements et de ses attributs

Dans cette section, il est question de la méthodologie permettant de répondre au premier sous-objectif, soit d'identifier les enregistrements et ses attributs dans une page Web. Pour ce faire, la première étape consiste à transformer un *graphe_HTML* en une séquence de jetons (section 3.2.1). Par la suite, un modèle utilisant les couches de type champs aléatoire conditionnel (voir section 2.2.3) est utilisé afin d'annoter ces séquences de jetons (section 3.2.2). La sortie du modèle sera finalement évaluée selon les métriques définies à la section 3.2.3.

3.2.1 Mise en jetons et annotation

Afin de procéder à l'annotation de séquences, les données en entrées doivent être une série continue de jetons (tokens). De ce fait, la première étape consiste à transformer la structure des *graphe_HTML* en une séquence continue. Ce procédé est illustré à la figure 3.5 et se divise en trois parties :

1. Tout d'abord, il faut parcourir le *graphe_HTML* et transformer celui-ci en séquence de balises HTML et de textes. Pour ce faire, l'arbre est parcouru selon un algorithme de parcours en profondeur (pre-order) afin de respecter la relation parent-enfant dans la séquence finale (figure 3.5b).
2. Par la suite, cette séquence est transformée en séquence continue de jetons. De ce fait, deux stratégies de mise en jeton (tokenization) sont nécessaires, soient une pour les balises HTML et une pour le texte brut (figure 3.5c) :
    a. Balise HTML : Afin de respecter la relation parent-enfant, les balises HTML sont disposées de façon à encapsuler ses enfants. la balise située au début est composée de l'étiquette «start» suivi du nom de la balise. De même, celle située à la fin est précédée de l'étiquette «end».
    b. Texte : Dû à la nature du texte contenu sur les forums, un algorithme de mise en jetons par codage de paire d'octets (BPE) sera entraîné. Celui-ci est une adaptation de l'algorithme BPE afin de segmenter des mots. Cette technique a été introduite par

Sennrich *et al.* (2016) afin de mieux représenter les mots rares et inconnus dans des tâches de traduction utilisant l'apprentissage machine (Sennrich *et al.*, 2016) (voir section 2.2.2).

3. Finalement, la dernière étape consiste à associer chaque jeton à son identifiant. Pour ce faire, une annotation de type BIO est utilisée (visible à la figure 3.5d) (voir section 2.2.3). Contrairement aux autres attributs, l'attribut enregistrement est annoté sur les jetons de type balises HTML.

3.2.2 Annotation de séquences

Le modèle choisi afin de procéder à l'annotation de séquences est le BiLSTM-CRF (Huang, Xu & Yu, 2015), choisi pour ses performances. Ce modèle permet de combiner une couche de champ aléatoire conditionnel, qui permet de capturer la relation entre les attributs à l'aide d'une matrice de transition (voir section 2.2.3), avec un «Bidirectional Long Short Term Memory» qui capture les relations sémantiques sur le long terme. L'implémentation de ce modèle est basée sur le répertoire Github «jidasheng/BiLSTM-crf» (https://github.com/jidasheng/BiLSTM-crf) (jidasheng, 2019).

Les hyperparamètres du modèle sont :

- Dimension du plongement de mots (Embedding dimension) (DP) : Correspond à la taille du vecteur représentant chaque mot du vocabulaire.
- Nombre de couches du BiLSTM (CB) : Correspond au nombre de couche de cellules BiLSTM.
- Nombre de cellules LSTM (HD) : Correspond au nombre de cellule LSTM par couche de cellules BiLSTM.
- Taux d'apprentissage (TA)
- Décroissance de poids (Weight decay) (DA)

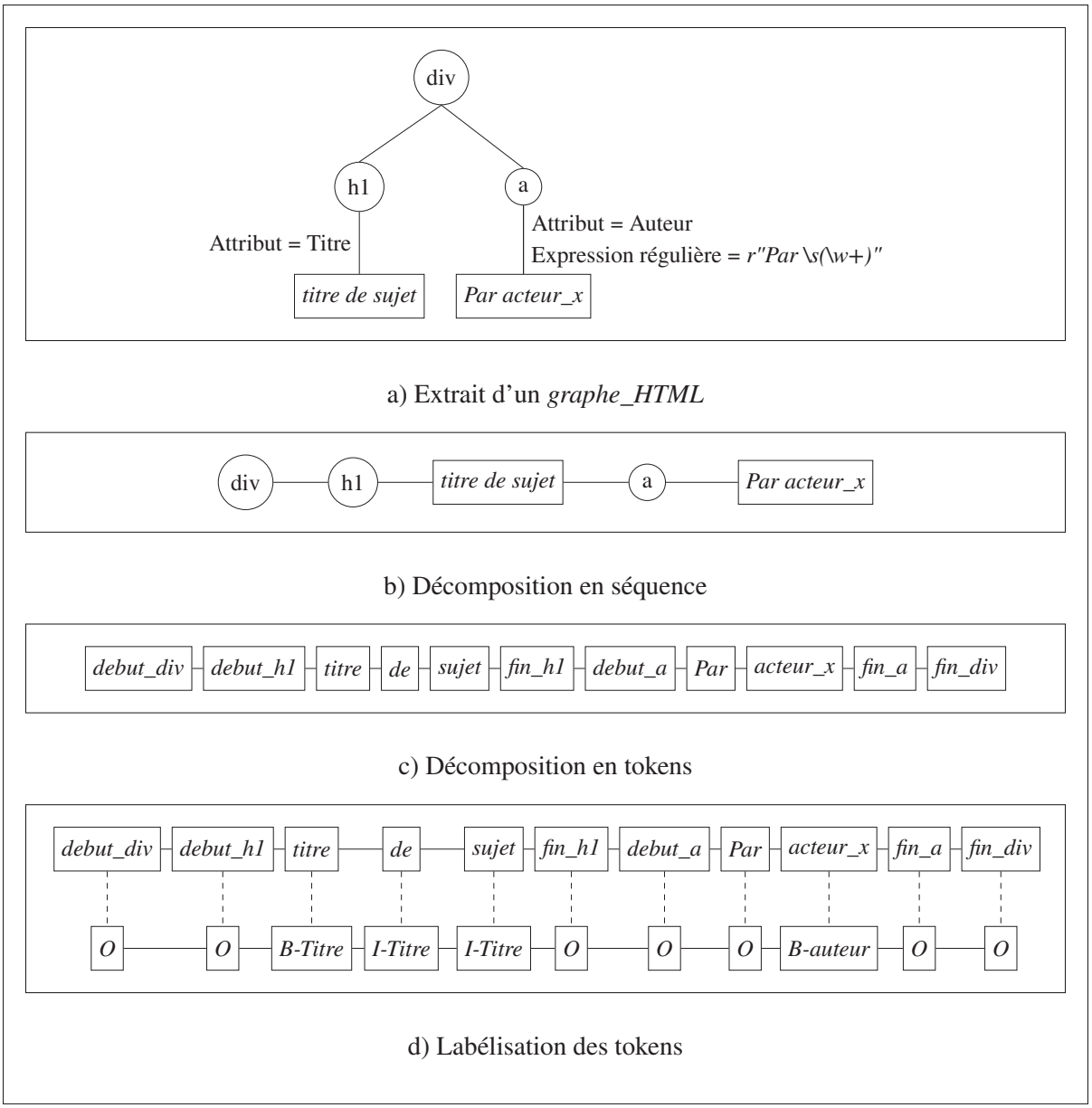


Figure 3.5 Représentation d'un fragment de *graphe_HTML* de sa forme en graphe vers la forme libellé

Lors de l'expérience multisource classique (voir section 4.3.1), ces hyperparamètres sont testés avec différentes valeurs afin de déterminer la meilleure combinaison afin d'obtenir le meilleur score macro-f1 selon les critères énumérés la section 3.2.3. Les valeurs choisis sont près des valeurs originales utilisées par Huang *et al.* (2015) (voir tableau 4.3).

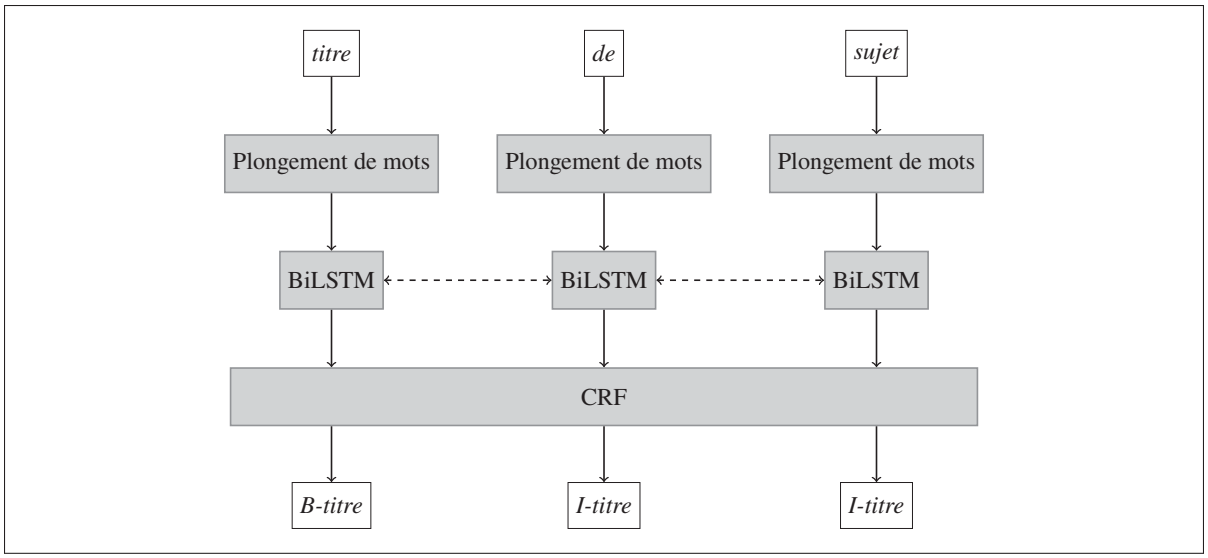


Figure 3.6 Architecture du BiLSTM-CRF

Lors de l'entraînement, le modèle utilise la fonction de perte de la couche CRF afin d'ajuster ses paramètres internes (poids). Afin de calculer cette fonction, le modèle utilise le logarithme de la probabilité d'obtenir le bon chemin (celui fournit par le jeu d'entraînement) divisé par la somme de la probabilités de tous les autres chemins possibles. Cette fonction tend alors vers zéro lorsque le modèle minimise les erreurs générées.

$$\text{Fonction de perte} \quad = \quad \log \frac{P_{reelle}}{P_1+P_2+\ldots+P_N}$$

3.2.3 Évaluation de l'identification des attributs

Lors de l'entraînement, le modèle servant à l'annotation de séquences est optimisé avec la fonction de coût de la couche CRF. Cette mesure tend à diminuer quand les prédictions générées par la couche CRF se rapprochent des données de référence. Afin de vérifier que cette mesure tend à améliorer les performances du modèle, trois autres mesures sont surveillées durant l'entraînement, soient la précision, le rappel ainsi que le F1, et ce pour chacun des attributs. De

plus, le macro F1 est calculé (sur toute les classes incluant "O") afin de déterminer la moyenne des F1, sans tenir compte de la pondération de chaque attributs.

$$\begin{aligned}
\text{Précision} &= \frac{TP}{TP+FP} \\
\text{Rappel} &= \frac{TP}{TP+FN} \\
\text{F1-pointage} &= \frac{2*Precision*Recall}{Precision+Recall} \\
\text{Macro F1-pointage} &= \frac{1}{N}\sum_{i=0}^{N} \text{F1-pointage}_i \text{ ou N est le nombre de classes}
\end{aligned}$$

Cette évaluation se fera entre les valeurs prédites et de référence de chaque jeton. De plus, deux types de mesures sont introduits, soient stricts et souples. La méthode stricte requiert que les jetons prédits et vrais soient les mêmes, tandis que la méthode souple ne prend pas en compte les préfixes de début (B) et interne (I) de l'annotation BIO.

L'évaluation en mode souple est introduite puisqu'il n'est pas nécessaire, lors du processus d'extraction des enregistrements et attributs, que chaque séquence étiquetée par le modèle soit une séquence BIO valide (soit qu'un ou une suite de I soit précédé par un B.). Par conséquent, il est possible qu'un modèle produise des séquences BIO invalides, mais dont la valeur de l'attribut prédit est la bonne.

Dans la figure 3.7, il est possible de voir l'effet d'une évaluation en mode stricte et souple. En mode stricte, des prédictions sont identifiées comme étant des erreurs puisqu'elles n'ont pas le bon préfixe (B ou I), même si l'attribut est le bon.

3.3 Extraction des enregistrements et de ses attributs

Dans cette section, il est question de se servir de la sortie du modèle, soit une séquence BIO, afin d'extraire les enregistrements et ses attributs de la page Web associée à la séquence. Cela correspond au sous-objectif 2 de cette recherche.

La première étape est de reconstruire le *graphe_HTML* à partir de la séquence BIO (secion 3.3.1). Par la suite, chaque noeud du *graphe_HTML* est associé à un attribut à partir de la séquence BIO

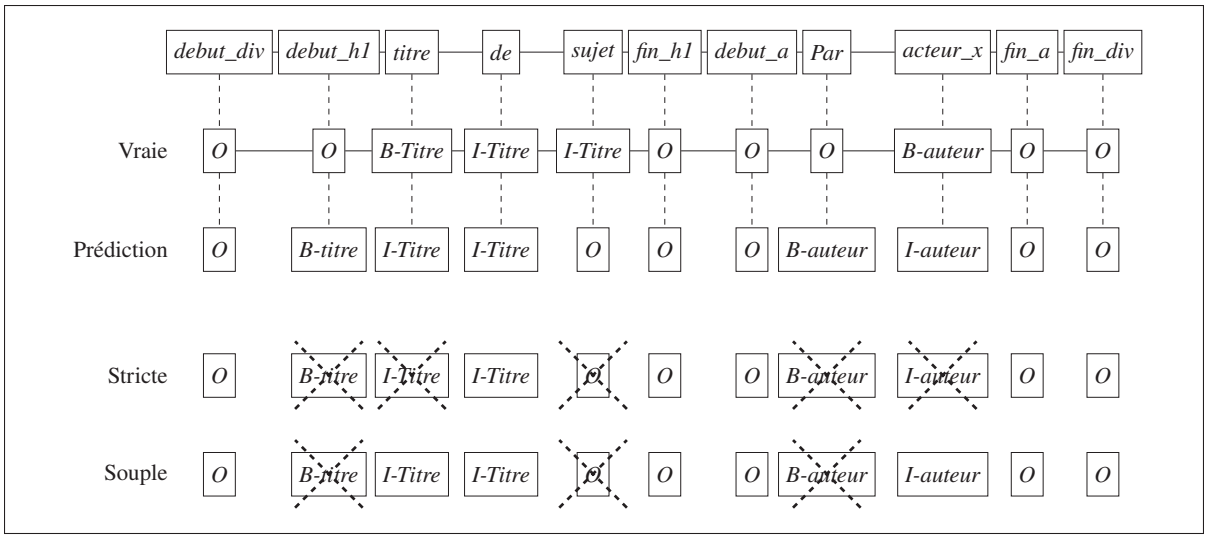


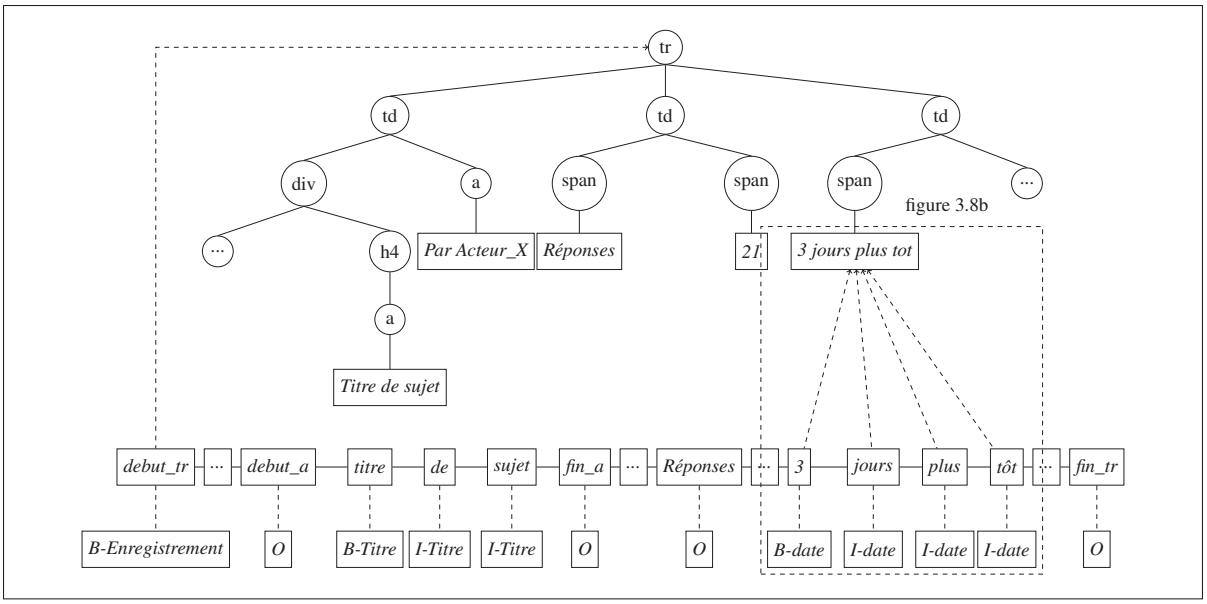
Figure 3.7 Représentation des deux méthodes pour évaluer les performances, soit strict et souple.

qui lui est associée (section 3.3.2). Finalement, le *graphe_HTML* est parcouru afin d'extraire les enregistrements (section 3.3.3) et leurs attributs (section 3.3.4). La figure 3.8 représente ce processus.

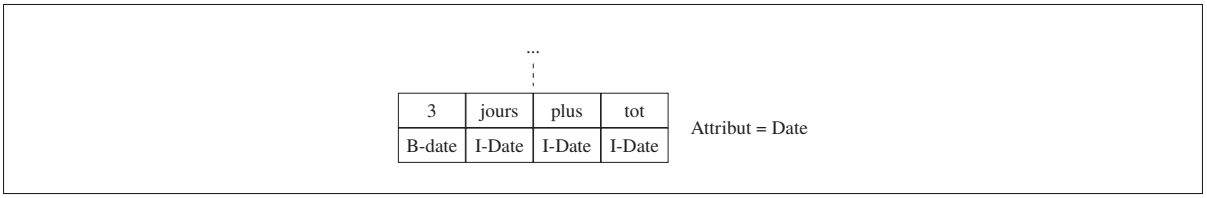
Suite à l'extraction des enregistrements et leurs attributs, la performance de la solution est mesurée en comparant la valeur prédite de chaque noeud avec la valeur réelle provenant des règles d'extractions manuelles.

3.3.1 Reconstruction de la structure de données

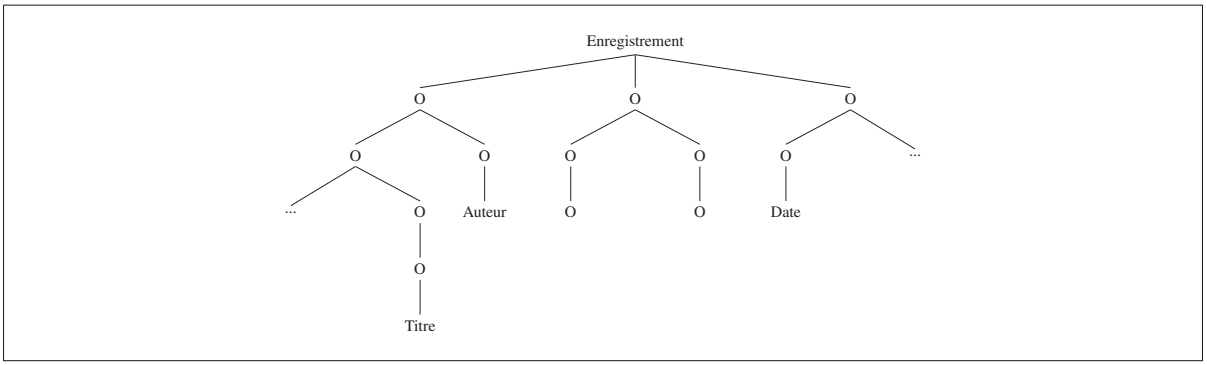
Afin de procéder à l'extraction des enregistrements et de ses attributs à partir d'une séquence BIO, il faut reconstruire le *graphe_HTML* duquel la séquence BIO a été extraite. Cette étape permet d'obtenir un *graphe_HTML* dont chaque noeud est associé à une sous-séquence BIO. Ce procédé est illustré à la figure 3.8a



a) Association des prédictions avec les noeuds du *graphe_HTML*



b) Un noeud reconstitué avec les prédictions



c) *graphe_HTML* dont les noeuds ont été prédits

Figure 3.8 Reconstruction d'un *graphe_HTML* à partir de la sortie du modèle (certaines lignes ne sont pas représentées afin d'alléger la figure)

3.3.2 Classification des noeuds du *graphe_HTML*

Dans le but d'associer un attribut à un noeud du *graphe_HTML*, un algorithme (*algorithme de pointage BIO*) a été conçu afin d'associer un pointage entre 0 et 1 à une séquence BIO. De ce fait, un noeud ne contenant aucun jeton étiqueté (seulement de "O") aura un pointage de zéro pour chaque attribut.

Pour calculer le pointage, l'*algorithme de pointage BIO* associe chaque transition à un pointage de 1 ou 0 et divise la somme des transitions par le nombre de transitions. Étant donné que la version souple ne prend pas en compte les *B-* et *I-*, une transition débutant par *I-Titre* est valide. Le tableau 3.3 présente le pointage associé à une séquence BIO pour l'attribut titre.

Tableau 3.3 Sortie de l'algorithme de pointage BIO pour l'attributs Titre selon des exemples de séquences BIO

Séquence BIO	Mode stricte	Mode souple
B-Titre - I-Titre - I-Titre - I-Titre	1	1
I-Titre - I-Titre - I-Titre - I-Titre	0	1
O - O - B-Titre - B-Titre	0,75	1
B-Titre - I-Titre - I-auteur - I-titre	0,5	0,75

Exemple de calcul en mode stricte pour la 4e ligne du tableau 3.3 :

$$\begin{aligned} \text{DÉBUT -> B-Titre -> I-Titre -> I-auteur -> I-titre} &= (1+1+0+0)/4 \\ &= 0,5 \end{aligned}$$

3.3.3 Extraction des enregistrements

Pour extraire chaque enregistrement, deux algorithmes ont été conçus en se basant sur les travaux de Ujwal *et al.* (2017). Ces deux algorithmes retournent chacun une série de noeuds qui sont susceptibles de contenir un enregistrement.

Les deux sous-sections suivantes présentent les deux algorithmes développés. Chacun d'eux repose sur la méthode *obtenir_attributs()* qui retourne un pointage associé à chaque attribut potentiel d'un noeud donné (voir section 3.3.2).

3.3.3.1 Algorithme 1 : Exploration en profondeur

Le premier est l'algorithme *Exploration en profondeur*. Celui-ci exploite le balisage BIO afin de trouver les parents dont les enfants ont été libellés comme étant un enregistrement (*B-Enregistrement*). Lors de son exécution, il parcourt le *graphe_HTML* en profondeur et, pour chaque noeud avec des enfants, il divise le nombre de noeuds étant un enregistrement par le nombre total d'enfants. Si ce ratio est plus grand qu'un certain niveau, les enfants sont ajoutés à la liste d'enregistrements potentiels. Par défaut, ce ratio est de 0. Donc tous les parents ayant en enfant qui a été libellé comme enregistrement est ajouté à la liste de parents potentiels.

Algorithme 3.1 Pseudo-code de l'algorithme exploration en profondeur

```
Require: G {Un graphe_HTML contenant une série de noeuds}
Require: potentiel_enregistrements = []
  for all n ∈ G do
    attributs_enfants = attributs_enfants(n.enfants)
    if enregistrement in attributs_enfants then
      if attributs_enfants[enregistrement] > minimum then
        potentiel_enregistrements.extend(n.enfants)
      end if
    end if
  end for
  return enregistrement_potentiels
```

3.3.3.2 Algorithme 2 : Parcours des feuilles

L'algorithme *Parcours des feuilles*, quant à lui, part du principe que chaque enregistrement aura au minimum un attribut commun. Dans ce cas-ci, chaque enregistrement doit contenir un titre. L'algorithme parcourt alors chaque feuille du *graphe_HTML* afin de trouver des titres. Quand il en trouve un, il note chaque parent de celui-ci récursivement, soit jusqu'à temps qu'il tombe

sur un parent déjà noté. À la fin, les noeuds ayant été le plus souvent notés sont ceux qui sont susceptibles d'être des parents d'enregistrements.

Algorithme 3.2 Pseudo-code de l'algorithme parcours des feuilles

Require: $\mathcal{G}$ {Un *graphe_HTML* contenant une série de noeuds}
Require: $parent\_enregistrements = []$
Require: $parent\_potentiels = []$
Require: $enregistrement\_potentiels = []$
  for all Les feuilles $f \in \mathcal{G}$ **do**
    $attributs = obtenir\_attributs(f)$
    if $attributs == titre$ **then**
      $parent\_potentiels = ajouter\_parents(parent\_potentiels, f)$
    end if
  end for
  return $enregistrement\_potentiels$

3.3.4 Extraction des attributs

Afin d'extraire les attributs des enregistrements, un algorithme a été conçu (*Extraction des attributs*). Celui-ci reçoit un noeud provenant d'un *graphe_HTML* reconstruit (voir figure 3.8) qui est susceptible de contenir un enregistrement (voir section 3.3.3). L'algorithme parcourt chaque feuille de ce noeud, et pour chaque attribut recherché, la feuille qui a le plus haut pointage est la gagnante.Ainsi, la sortie de cet algorithme est une liste de feuilles du *graphe_HTML* associé à un attribut.

Finalement, les trois algorithmes présentés précédemment sont combinés (algorithme 3.2, 3.1 et 3.3) dans l'algorithme *Extraction enregistrements*. Tout d'abord, la recherche d'enregistrement débute avec l'algorithme *Exploration en profondeur*. Si aucun enregistrement n'est trouvé, l'algorithme *Parcours des feuilles* est alors utilisé. Suite à cela, si des enregistrements potentiels ont été trouvés, l'algorithme *Extraction des attributs* est appelé sur chacun d'eux afin de trouver les attributs des enregistrements. La sortie est alors une série d'enregistrement, chacun associé à une série d'attributs.

Algorithme 3.3 Pseudo-code de l'algorithme extraction des attributs

Require: $\mathcal{N}$ {Un noeud d'un *graphe_HTML* contenant potentiellement un enregistrement}
Require: $attributs\_recherches = []$
Require: $attributs\_trouves = []$
  for all Les feuilles $f \in \mathcal{N}$ **do**
    $attribut = obtenir\_attributs(f)$
    if $attribut \in attributs\_recherchesandattribut.score >$
    $attributs\_trouves[attribut].score$ **then**
      $attributs\_trouves[attribut] = attribut$
    end if
  end for
  return $attributs\_trouves$

3.3.5 Évaluation de l'extraction

Afin d'évaluer les performances d'extraction, la sortie de l'algorithme *Extraction enregistrements* est utilisée. On compare alors l'attribut de référence de chaque noeud du *graphe_HTML* avec la valeur prédite. Les mesures de rappel, précision, F1 et macro-F1 (décrit à la section 3.2.3) seront appliquées pour l'analyse des résultats. Cette évaluation permet de déterminer si la sortie du modèle (voir section 3.2) a permis aux algorithmes d'extraire les bons enregistrements, et les bons attributs associés à chacun d'eux.

CHAPITRE 4

EXPÉRIMENTATIONS

Dans ce chapitre, il est question des expérimentations effectuées afin de tester la solution dans un contexte multisource classique et fractionnaire, et ainsi, déterminer si l'approche proposée présentée dans le chapitre précédent permet de répondre aux objectifs de cette recherche. Ainsi, la section 4.1 fait le point sur les données recueillies et traitées, tandis que la section 4.2 fait le point sur la mise en jetons. Suite à cela, trois expériences permettant de tester la solution sont décrites dans la section 4.3.

4.1 Ensemble de données

Comme mentionné dans la section 3.1.3, deux ensembles de données ont été créés, soit l'ensemble A et B. L'ensemble A est utilisé afin de tester la solution dans un contexte multisource classique, tandis que l'ensemble B est utilisé dans un contexte multiource fractionnaire.

Les données contenues dans les deux ensembles sont annotées à partir des règles d'extraction manuelle, telles que mentionnées dans la section 3.1.2. Celles-ci sont par la suite structurées en *graphe_HTML*, avant d'être validées et nettoyées (voir les sections 3.1.5, 3.1.4 et 3.1.6).

Le processus de validation a permis de détecter de nombreuses erreurs dans les règles d'extractions écrites par les opérateurs. Par conséquent, certaines règles ont dû être réécrites. Les erreurs encourues, tel la variation de la structure interne des pages Web, ont été discutées dans la section 2.1.1

Suite aux traitements des données, celles-ci sont divisées afin de créer l'ensemble A selon les critères établis dans la section 3.2. Le tableau 4.1 résume la division des sources. La division des sources pour l'ensemble A est disponible en annexe, au tableau I-1. Il est à noter que certaines sources ont plusieurs langues. De plus, les sources ayant le plus d'échantillons ont été privilégiées pour le jeu d'entraînement.

Tableau 4.1 Résumé de la dvision des sources

	Entrainement	Validation	Test
Nombre de sources	22	4	3
Langues	Anglais :15 Russe : 7 Autres : 2	Anglais : 3 Russe : 1	Anglais : 3 Russe : 1
Structure	Division : 11 Liste : 4 Tableau : 7	Division : 2 Liste : 0 Tableau : 2	Division : 1 Liste : 1 Tableau : 1
Présence de l'attribut «Date»	Oui : 5 Non : 17	Oui : 2 Non : 2	Oui : 2 Non : 1

4.2 Mise en jetons

L'analyse du corpus d'apprentissage BPE a permis de constater que celui-ci comprend un total de 3 212 caractères uniques. Un algorithme de mise en jeton a été entraîné pour une taille de vocabulaire de 5 000, 10 000, 50 000 et 100 000. Chaque taille de vocabulaire est utilisée lors de l'expérience consistant à évaluer l'effet de la taille du vocabulaire (section 4.3.2). Pour chacun des algorithmes de mise en jeton, un paramètre de recouvrement de 99% est utilisé. Par conséquent, 3096 caractères, sur un total de 3212, n'ont pas été retenu lors du processus de BPE, dû à leur faible occurrence.

Le tableau 4.2 compare la mise en jeton de trois séquences selon la taille du vocabulaire utilisé.

4.3 Évaluation

Afin d'évaluer la solution, soit l'identification et l'extraction des enregistrements et de leurs attributs, trois expériences sont effectuées. La première expérience (section 4.3.1) est de type multisource classique. Elle consiste en une recherche par quadrillage effectuée sur l'ensemble A. Par la suite, la combinaison de valeurs des hyperparamètres ayant donné les meilleurs résultats est utilisée dans la deuxième expérience (section 4.3.2) afin de déterminer la taille optimale de vocabulaire. Finalement, l'ensemble d'hyperparamètres aillant le mieux performé lors de la première expérience est utilisé avec la taille de vocabulaire ayant donné les meilleurs résultats

Tableau 4.2 Comparaison des différentes tailles de vocabulaire

Séquence	Taille du vocabulaire			
	5000	10000	50000	100000
What do u think about Fortnite	['-What', '-do', '-u', '-th', 'ink', '-about', '-Fortnite']	['-What', '-do', '-u', '-think', '-about', '-Fortnite']	['-What', '-do', '-u', '-think', '-about', '-Fortnite']	['-What', '-do', '-u', '-think', '-about', '-Fortnite']
1HackYou	['-1', 'H', 'ack', 'You']	['-1', 'H', 'ack', 'You']	['-1', 'Hack', 'You']	['-1', 'Hack', 'You']
1 year ago	['-1', '-year', '-ago']	['-1', '-year', '-ago']	['-1', '-year', '-ago']	['-1', '-year', '-ago']

lors d'une expérience dans un contexte multisource fractionnaire avec l'ensemble B (section 4.3.3).

4.3.1 Évaluation de la solution dans un contexte multisource classique

La première expérience consiste à utiliser un algorithme de recherche par quadrillage afin de déterminer quels sont les meilleurs hyperparamètres du modèle décrit à la section 3.2.2. De ce fait, chacune des combinaisons du tableau 4.3 est testée. L'intervalle choisi se rapproche des valeurs par défaut du papier orignal de Huang *et al.* (2015).

Pour cette expérience, la taille de vocabulaire utilisé est de 5000 et l'ensemble de données A est utilisé.

Afin d'évaluer les performances du modèle, les métriques décrites dans la section 3.2.3 et la valeur de retour de la fonction de coût de la couche CRF sont mesurées sur le jeu d'entraînement et de validation lors de l'entraînement à la fin de chaque époque. Chaque modèle est entraîné

durant 10 époques (valeur déterminée suite à des tests préliminaires) et le modèle est enregistré lorsque la perte sur les données de validation est la plus faible.

Tableau 4.3 Valeur des hyperparamètres

Hyperparamètre	Valeurs
Dimension du plongement de mots (DP)	16, 64, 100, 300
Nombre de couches de BiLSTM (CB)	1,2
Nombre de cellules LSTM (HD)	64, 128
Taux d'apprentissage (TA)	0.01, 0.001
Décroissance de poids (DA)	0, 0.1

Suite à l'entraînement, chaque modèle est testé sur le jeu de test. Lors de cette étape, la méthode d'évaluation présentée à la section 3.3.5 est utilisée sur les *graphe_HTML* qui auront été annotés avec le modèle afin de mesurer les performances. Les métriques décrites à la section 3.3.5 est utilisées. De plus, l'algorithme de pointage est configuré en mode stricte afin d'évaluer le modèle dans les conditions les plus strictes possibles (voir section 3.3.2).

4.3.2 Évaluation de l'effet de la taille du vocabulaire

Le but de cette expérience est de déterminer la taille optimale pour le vocabulaire. Pour ce faire, la meilleur combinaison d'hyperparamètres obtenues lors de l'expérience 4.3.1 est utilisée. Lors de cette expérience, une taille de vocabulaire différente de 5 000 est testée, soient 10 000, 50 000 et 100 000. Ces valeurs ont été choisies afin de garder le modèle léger (taille de vocabulaire de moins de 100 000) et suite aux bonnes performances de tests préliminaires effectués avec une taille de vocabulaire de 25 000.

L'évaluation est effectuée avec les mêmes mesures de performances que lors de l'expérience 4.3.1 avec l'ensemble de données A.

4.3.3 Évaluation de la solution dans un contexte multisource fractionnaire

Cette expérience a pour but de déterminer les performances du modèle dans un scénario multisource fractionnaire. De ce fait, l'ensemble B est utilisé ainsi que le modèle ayant obtenu les meilleurs résultats lors de l'expérience 4.3.2. L'évaluation sera effectuée sur le jeu de test.

CHAPITRE 5

RÉSULTATS

Dans ce chapitre, les résultats des trois expériences décrites au chapitre précédent 4.3 est présentés. La section 5.1 présente les résultats suite à l'expérience de recherche des meilleurs hyperparamètres pour le modèle. Par la suite, il est question, dans la section 5.2, de déterminer si la taille du vocabulaire influence les résultats. Suit l'expérience de type multisource fractionnaire dans la section 5.3. Finalement, les performances de la solution est présentées dans la section 5.4.

5.1 Expérience 1 : Recherche des meilleurs hyperparamètres pour le modèle dans un contexte multisource classique

Le tableau 5.1 classe les résultats des cinq meilleurs modèles selon la valeur de la macro F1 obtenue sur les données de test. Le macro F1 est calculée sur les noeuds du *graphe_HTML* tel que stipulé à la section 3.3.5. Chaque modèle testé est enregistré durant l'entraînement lorsque la valeur de est était la plus basse sur les données de validation. Les résultats complets sont disponibles dans l'annexe 1.1. La moyenne des différentes macro F1 sur le jeu de test est de 0,778, tandis que la moyenne de la perte sur les données de validation est de 246,32.

Tableau 5.1 5 meilleurs résultats selon le macro F1 obtenus sur les données de test lors de l'expérience de recherche des meilleurs hyperparamètres

	Hyperparamètres						
ID	**DP**	**CB**	**HD**	**TA**	**DA**	**Perte (Validation)**	**Macro F1 (Test)**
1	100	2	64	0.01	0.1	120.15	0.995
2	16	2	64	0.01	0.1	128,14	0.989
3	300	1	64	0.001	0	339.60	0.972
4	300	1	64	0.01	0.1	122.76	0.967
5	16	1	128	0.01	0	200.79	0.957

Suite à la lecture de ces résultats, la combinaison d'hyperparamètres ayant obtenu les meilleurs résultats est présentée au tableau 5.2. Ces paramètres sont utilisés lors de l'expérience permettant de déterminer l'effet de la taille du vocabulaire utilisé.

Tableau 5.2 Hyperparamètres ayant généré les meilleurs résultats sur le jeu de test

Hyperparamètre	**Valeurs**
Dimension du plongement de mots (DP)	100
Nombre de couches de BiLSTM (CB)	2
Nombre de cellules LSTM (HD)	64
Taux d'apprentissage (TA)	0.01
Décroissance de poids (DA)	0.1

Le graphique 5.1 illustre la progression de la perte, mesurée sur le jeu d'entraînement et de validation à la fin de chaque époque, pour les cinq modèles ayant obtenu les meilleurs résultats sur les données de test. En analysant les résultats, il est possible de constater que les deux modèles ayant le mieux performé sur les données de test, ainsi que le 5e, ont atteint leur perte minimale sur les données de validation avant la fin de la 2e époque. Les autres (3 et 4) ont atteint ce minimum à la fin de la 8e époque.

La matrice de confusion présentée à la figure 5.2 présente les résultats du modèle 1 en mode strict (voir section 3.2.3) sur le jeu de validation. Il est alors possible de constater que le modèle est en mesure de bien étiqueter les titres et les dates, mais il a plus de difficulté au niveau des auteurs. Ce constat s'observe aussi sur la figure 5.3, qui met en évidence la précision, le rappel et le F1 de chacun des attributs du modèle 1 en mode souple.

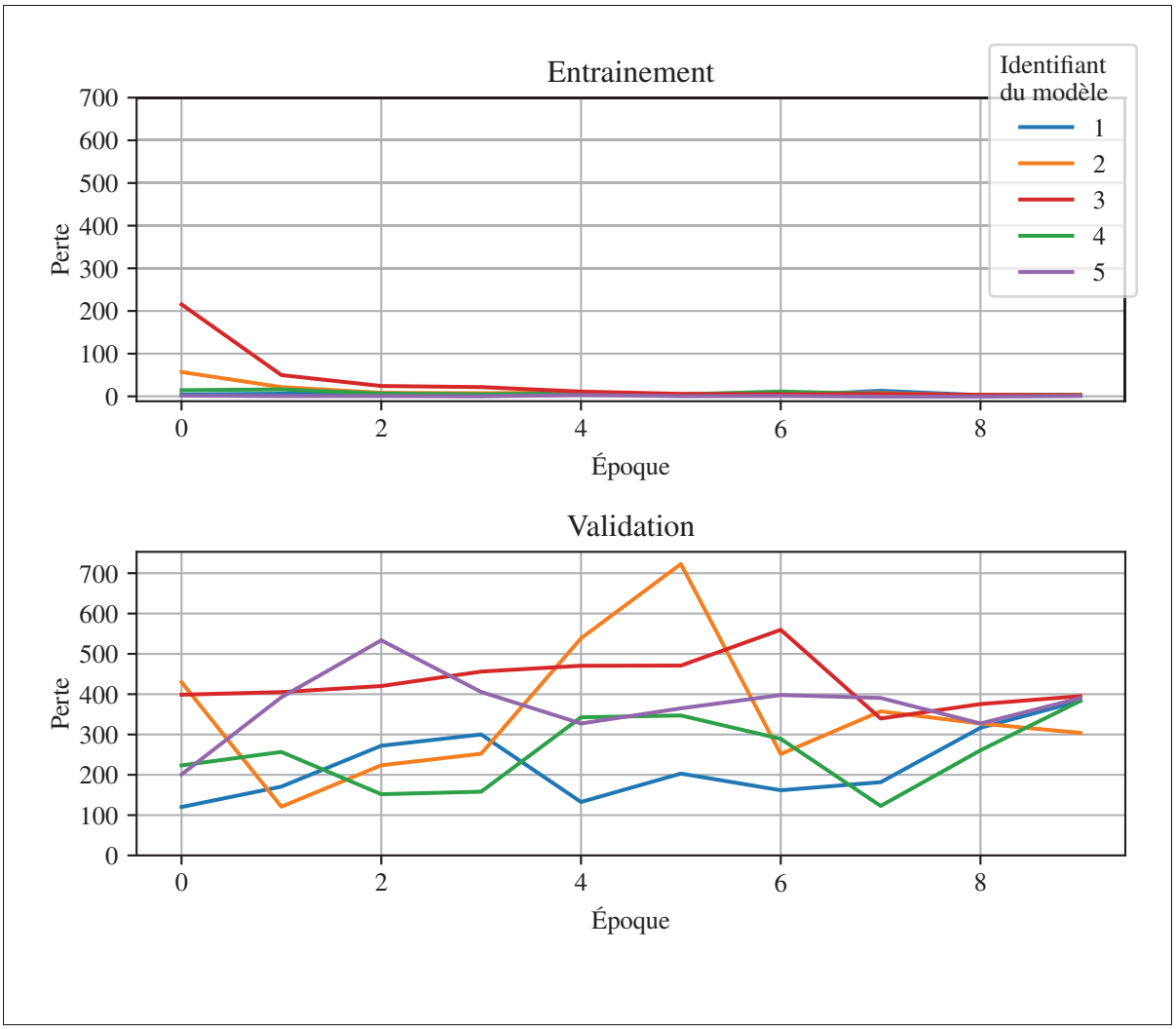


Figure 5.1 Perte sur les données d'entraînement et de validation sur l'ensemble A lors de l'expérience 1

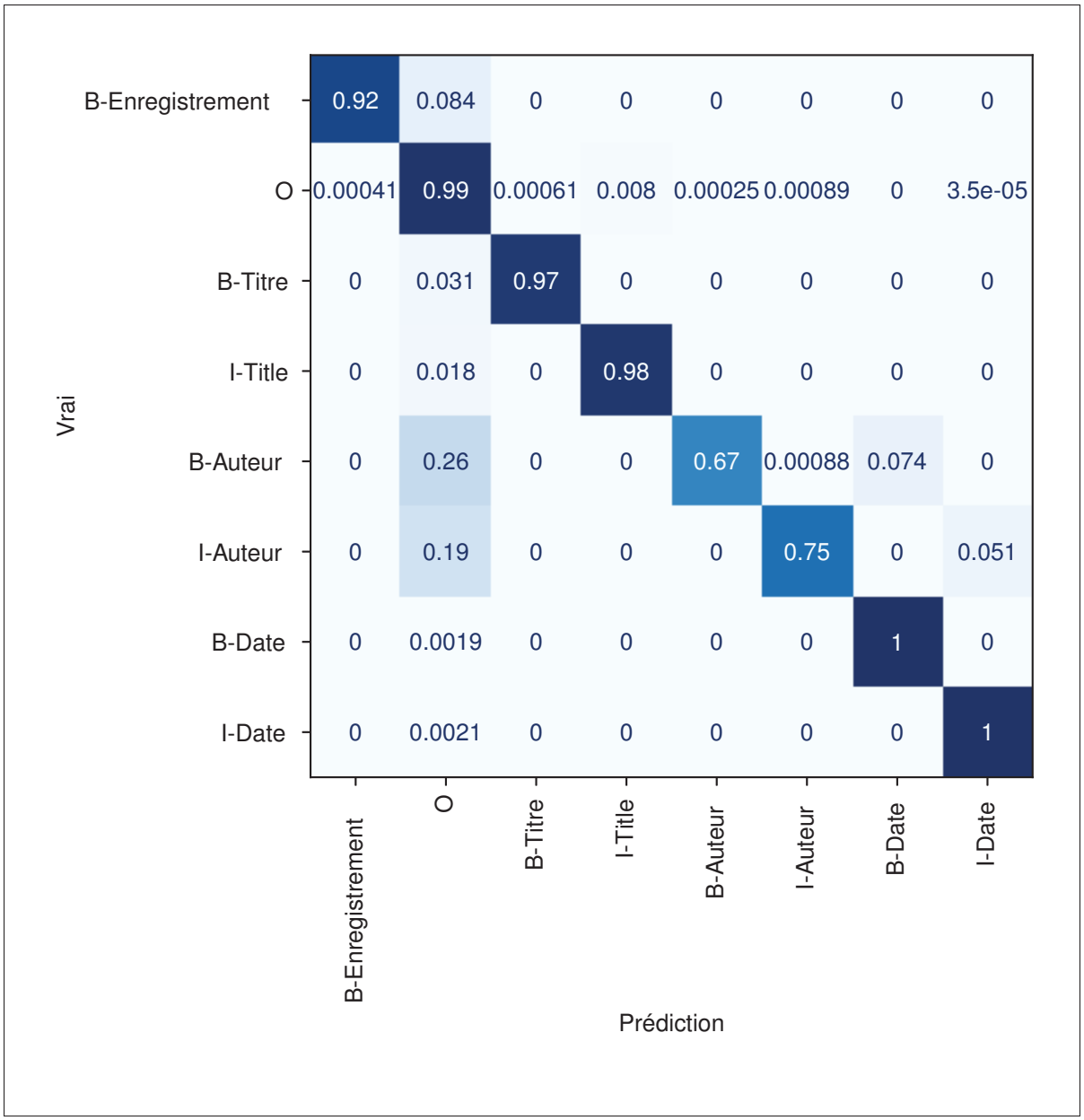


Figure 5.2 Matrice de confusion du jeu de validation sur le modèle 1 de l'expérience 1

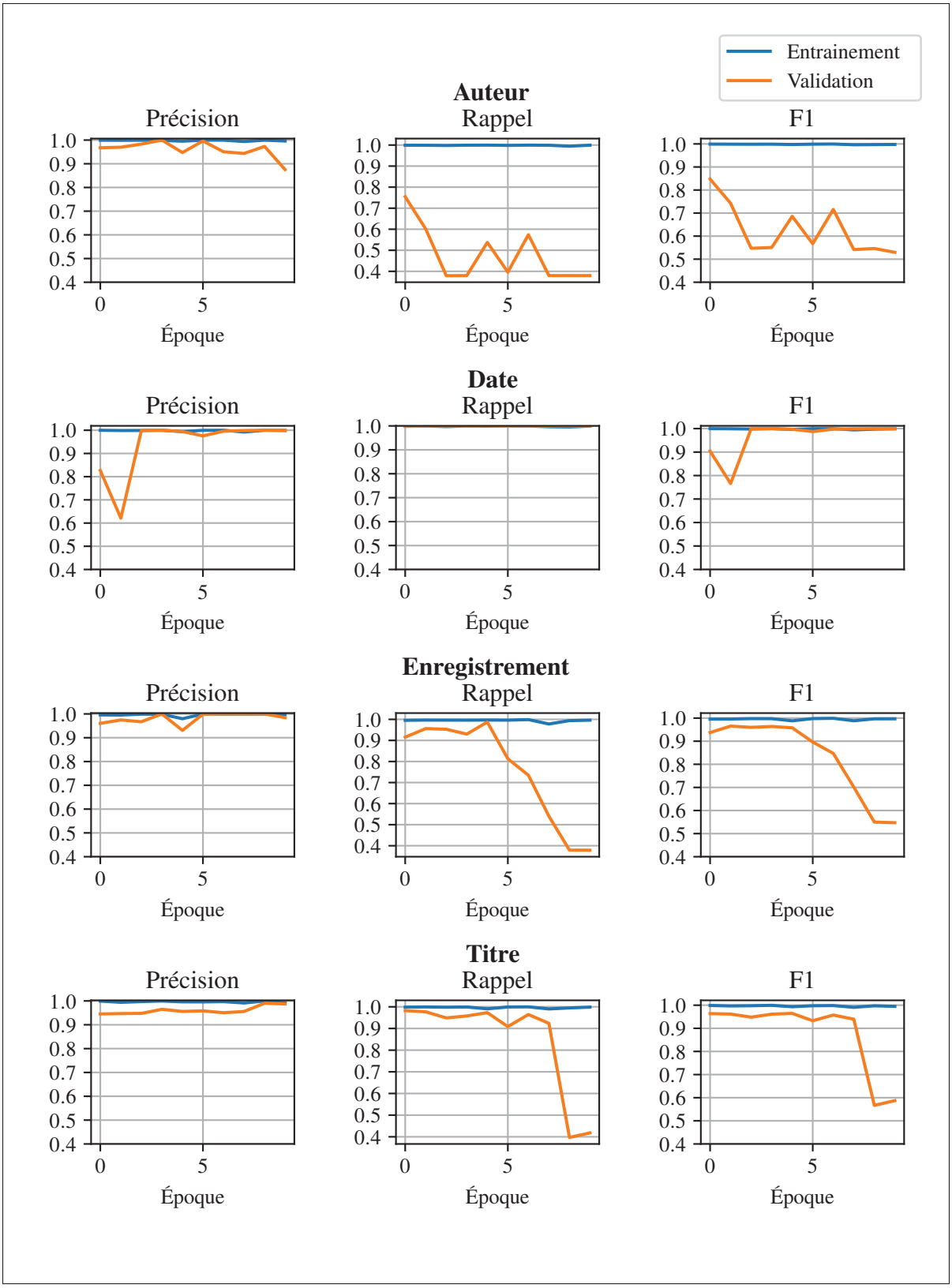


Figure 5.3 Précision, rappel et F1 des différents attributs en mode souple

5.2 Expérience 2 : Effet de la taille du vocabulaire

Cette expérience consiste à déterminer si la taille du vocabulaire a un effet sur les résultats du modèle. De ce fait, les hyperparamètres du tableau 5.2 ont été utilisés avec différentes tailles de vocabulaire. Comme lors de l'expérience précédente, l'ensemble A est utilisé et le modèle est évalué sur les données de validation à la fin de chaque époque. Le modèle enregistré à l'époque générant le moins de perte sur les données de validation est utilisé afin de procéder aux mesures sur le jeu de test.

Le tableau 5.3 illustre les résultats de chaque modèle selon la taille du vocabulaire utilisé. Les résultats présentés dans la première rangée, soit avec la taille de vocabulaire de 5 000, proviennent de l'expérience précédente (voir section 5.1) [1].

Tableau 5.3 Macro F1 obtenues sur les données de test selon la taille du vocabulaire

Taille du vocabulaire	Perte (validation)	Macro F1 (test)
5000	Moyenne de 246,32	Moyenne de 0,778
10000	149.20	0.790
50000	84.16	0.744
100000	58.36	0.808

Le graphique 5.4 illustre la progression de la perte sur le jeu d'entraînement et de validation selon la taille du vocabulaire. Il est possible de constater que les modèles ayant une taille de vocabulaire plus élevée (50 000 et 100 000) atteignent leur minimum plus tard (époque 10 et 7 respectivement) sur le jeu de validation, comparativement aux modèles ayant une taille de vocabulaire moins élevée (époque 1 pour 5000 et époque 2 pour 10 000).

Suite à l'analyse de ses résultats, il est possible de constater que la moyenne de la macro F1 sur le jeu de tests, si l'on ne considère pas le modèle avec une taille de vocabulaire de 5000, est 0.779. Cette valeur est donc près de la moyenne observée lors de l'expérience précédente sur une taille de vocabulaire fixée à 5 000. De plus, il est intéressant de constater que la perte sur les

[1] En raison du temps engendré par chaque expérience (environ 7h par combinaisons testées), une seule expérience a été réalisée afin de tester les autres tailles.

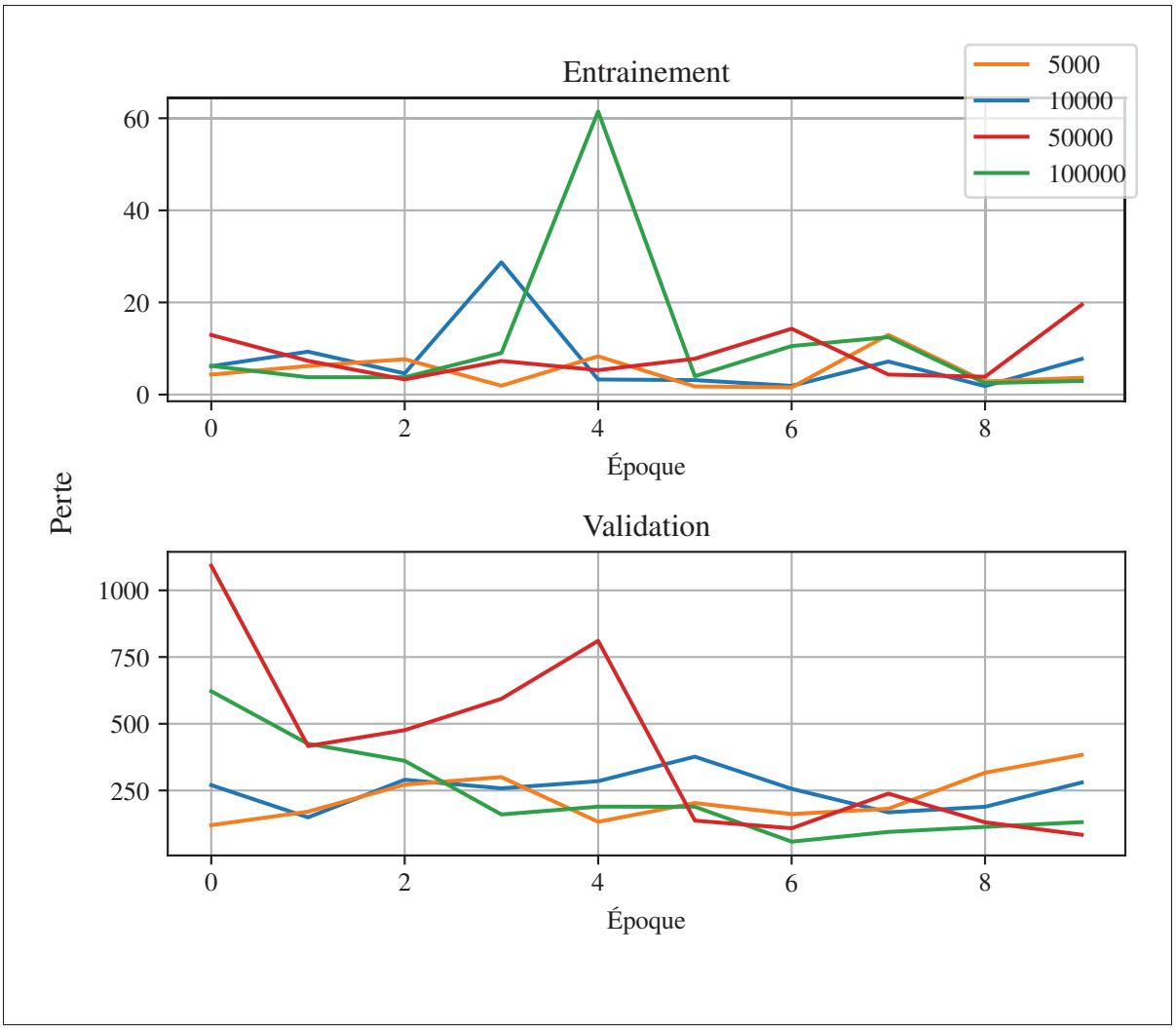


Figure 5.4 Perte sur les données d'entraînement et de validation sur l'ensemble A lors de l'expérience 2

données de validation tend à diminuer plus la taille du vocabulaire augmente. En effet, si l'on compare les pertes obtenues sur le jeu de validation sur nouvelle taille de vocabulaire testée, celles-ci sont toutes inférieures à la moyenne obtenue lors de l'expérience précédente (246,32).

5.3 Expérience 3 : Validation de la solution dans un contexte multisource fractionnaire

La dernière expérience a pour but de déterminer si le modèle est en mesure de bien performer dans un scénario multisource fractionnaire. De ce fait, l'ensemble B est utilisé avec les hyperparamètres du tableau 5.2 et une taille de vocabulaire de 5000.

Le tableau 5.4 contient la macro F1 calculée sur le jeu de test de l'ensemble B et ce pour chacune des sources de l'ensemble.

Suite aux résultats, il est intéressant de constater que la solution est en mesure, pour toutes les sources, d'obtenir une F1 test plus élevée que 97% sur le jeu de test. De plus, le modèle est en mesure de bien performer sur les sources ayant un échantillonnage élevé, en plus de celles ayant un échantillonnage faible, et ce peu importe la langue du forum.

5.4 Analyse des performances

Dans cette section, il sera question de l'analyse des performances de la solution développée. Par conséquent, la section 5.4.1 fait état du temps d'exécution entre la solution développée et celle reposant sur l'écriture et la maintenance de règles d'extraction. Par la suite, dans la section 5.4.2, il est question d'analyser la sortie du modèle combiné aux algorithmes d'extractions.

5.4.1 Temps d'exécution

Une des contraintes de ce mémoire est de proposer une méthode ayant un faible temps d'exécution lors de l'extraction. De ce fait, le tableau 5.5 compare le temps d'exécution moyen de la solution comparativement à un système manuel. Le système en question est celui de la compagnie Flare Systems, et repose sur l'écriture de règles d'extractions (voir section 2.1.1)

Le système automatique prend en moyenne environ 46% de temps supplémentaire comparativement au système utilisant des règles écrites manuellement. Ce temps ne prend cependant pas en compte le temps de configuration et de maintenance des règles manuelles. En effet, un opérateur humain est nécessaire afin d'écrire et maintenir celles-ci (voir section 2.1), tandis que

Tableau 5.4 Macro F1 sur le jeu de test de l'ensemble B
suite à l'expérience multisource fractionnaire

Source	**Échantillons**	**Langue**	**Structure**	**Macro F1 (Test)**
S1	596	Anglais	Division	0,99
S2	1200	Russe	Division	1
S3	1200	Russe	Division	0,97
S4	1079	Russe	Division	1
S5	739	Anglais, Russe	Tableau	0,99
S6	1200	Anglais	Tableau	1
S7	1200	Anglais	Tableau	1
S8	1200	Anglais, Multiples	Liste	0,99
S9	1200	Anglais	Tableau	0,99
S10	1200	Anglais	Division	0,98
S11	895	Anglais	Division	1
S12	1200	Anglais	Division	0,99
S13	1200	Russe	Division	0,99
S14	484	Anglais, Russe	Liste	0,99
S15	68	Francais	Divison	0,99
S16	279	Anglais	Tableau	0,97
S17	1200	Anglais, Russe	Tableau	0,99
S18	130	Anglais	Tableau	0,99
S19	1200	Anglais	Division	0,99
S20	1200	Russe	Division	0,98
S21	86	Anglais	Division	0,99
S22	253	Russe	Tableau	0,97
S23	1200	Anglais	Liste	1
S24	1200	Anglais	Division	1
S25	1200	Anglais	Tableau	1
S26	290	Anglais	Division	0,99
S27	948	Anglais	Liste	0,98
S28	198	Francais , Anglais	Liste	1
S29	1200	Anglais	Tableau	0,99

le système automatique ne repose sur aucune configuration manuelle une fois l'entraînement effectué. De plus, très peu d'attention a été portée sur l'optimisation des performances de la solution automatique, comparativement à la solution manuelle, qui a fait l'objet de nombreux perfectionnements au cours des années.

Tableau 5.5 Comparaison du temps d'exécution des approches manulle et automatique

Algorithme	Temps d'exécution moyen (seconde)
Extraction règles manuelles	0,474
Extraction automatique	0,693

5.4.2 Sortie du modèle

Dans cette section, il est question d'examiner les sorties du modèle combinées aux algorithmes d'extraction. De ce fait, le tableau 5.6 présente la sortie d'un sujet de discussion provenant du jeu de test de l'ensemble A. Le modèle utilisé est le modèle ayant le mieux performé sur les données de test lors de l'expérience de recherche des hyperparamètres (identifiant 1). La séquence originale correspond à la séquence encapsulée par la balise HTML contenant l'attribut. Les jetons correspondent aux jetons associés à cette séquence et les étiquettes sont les valeurs réelles et prédites de chacun de ces jetons.

Tableau 5.6 Exemple de sortie du système automatique, sur une page Web du jeu de test de l'ensemble A, effectué avec le modèle 1 de l'expérience 1

Auteur	**Séquence originale**	BullyProof
	Jetons	['-Bul', 'ly', 'Pro', 'of']
	Étiquette (vraie)	['B-auteur', 'I-auteur', 'I-auteur', 'I-auteur']
	Étiquette (prédiction)	['B-auteur', 'I-auteur', 'I-auteur', 'I-auteur']

Date	**Séquence originale**	Feb 24, 2020
	Jetons	['-Feb', '-24,', '-2020']
	Étiquette (vraie)	'B-date', 'I-date', 'I-date'
	Étiquette (prédiction)	'B-date', 'I-date', 'I-date'

Titre	**Séquence originale**	Hackers Steal Crypto by Impersonating Top VPN Service
	Jetons	['-Hackers', '-St', 'eal', '-Crypt', 'o', '-by', '-Im', 'p', 'erson', 'ating', '-Top', '-VPN', '-Service']
	Étiquette (vraie)	['B-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre']
	Étiquette (prédiction)	['B-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre', 'I-titre']

Les résultats présentés au tableau 5.6 représentent bien la sortie du modèle sur le jeu de test de l'ensemble A. En effet, après l'analyse manuelle de nombreux échantillons, seules quelques erreurs ont été trouvées. Ces erreurs étaient marginales et étaient principalement composées de dates manquantes. Après leur analyse, la majorité de ces erreurs était due au fait que le modèle n'avait pas bien étiqueté le premier jeton de la séquence en omettant un "B-" (tableau 5.7). Comme l'algorithme d'extraction était configuré en mode strict, cela rend la séquence invalide et ne peut être considérée comme une date. Cependant, le mode souple aurait accepté cette séquence (voir section 3.3.2).

Tableau 5.7 Exemple d'erreur de sortie du système automatique, sur une page Web sur le jeu de test de l'ensemble A, effectué avec le modèle 1 de l'expérience 1

Date	**Séquence originale**	October 19, 2005
	Jetons	['-October', '-19', ',', '-200', '5']
	Étiquette (vraie)	['B-date', 'I-date', 'I-date', 'I-date', 'I-date']
	Étiquette (prédiction)	['O', 'O', 'I-date', 'I-date', 'I-date']

La métrique utilisée afin de déterminer le modèle le plus performant sur le jeu de test ne prend pas en compte le scénario où un attribut ne se limite pas aux bornes d'une balise HTML (voir section 2.3.1). Cependant, il est possible de visualiser le comportement du modèle dans ce type de scénario. Le tableau 5.8 illustre un exemple de sortie du système automatique sur une source dont la mention de l'auteur est bruitée. L'exemple en question provient de le jeu de validation de l'ensemble A, puisqu'aucun scénario de ce type n'est présent dans le jeu de test.

Les étiquettes associées aux jetons de la séquence bruitée du tableau 5.8 permettent de constater que le modèle a été en mesure de faire la distinction entre le nom d'utilisateur et le bruit ("by") autour de ce dernier. En effet, le jeton "by" a été étiqueté comme n'appartenant à aucune catégorie, tandis que les autres ('-B', 'ig', 'S', ...), ont bien été étiquetés comme faisant partie d'un auteur.

Tableau 5.8 Exemple de sortie du système automatique, sur une page Web du jeu de validation de l'ensemble A, effectué avec le modèle 1 de l'expérience 1 dont l'auteur est bruité

Auteur	**Séquence originale**	by BigSpice
	Jetons	['-by', '-B', 'ig', 'S', 'p', 'ice']
	Étiquette (vraie)	['O', 'B-auteur', 'I-auteur', 'I-auteur', 'I-auteur', 'I-auteur']
	Étiquette (prédiction)	['O', 'B-auteur', 'I-auteur', 'I-auteur', 'I-auteur', 'I-auteur']

CHAPITRE 6

DISCUSSION

L'objectif principal de ce mémoire était de développer une approche permettant l'extraction des sujets de discussions et leurs attributs respectifs (titre, auteur et date de publication). De plus, l'hypothèse était qu'il est possible d'utiliser des outils provenant du domaine du TALN, et plus particulièrement des outils d'annotation de séquences, afin de procéder à l'extraction de contenu sur le Web. De ce fait, deux sous-objectifs étaient poursuivis, soit l'identification des attributs dans une séquence composée de texte et balises HTML, et l'extraction de ceux-ci. Afin de répondre à ces objectifs, trois expériences ont été réalisées. Celles-ci étaient divisées en deux catégories, soit multisource classique et fractionnaire (voir section 4.3). Les résultats de ces expériences seront discutés dans les sections 6.1 et 6.2 respectivement. Par la suite, les performances de la solution dans un contexte industrielle seront discutées (section 6.5), ainsi que la contribution de ce travail de recherche (section 6.7).

6.1 Expériences multisources classiques

La première expérience réalisée consistait à déterminer les meilleures combinaisons d'hyperparamètres du modèle choisi, soit le BiLSTM-CRF dans un contexte multisource classique. Pour ce faire, l'ensemble de données A a été utilisé pour effectuer une recherche par quadrillage. De plus, la taille du vocabulaire a été fixée à 5000 pour toutes les combinaisons. Au final, 64 combinaisons ont été testées générant des macro F1 calculés sur le jeu de test entre 0,491 et 0,995 (moyenne de 0,778). Malgré les bons résultats obtenus avec la combinaison d'hyperparamètres ayant le mieux performés sur les données de test ((DP :100, CB :2, HD :64, TA :0,01, DA : 0.1), il est difficile d'affirmer que ceux-ci en sont la cause.

En effet, lors de la deuxième expérience, qui permettait de vérifier l'effet de la taille du vocabulaire, ces mêmes paramètres ont été utilisés avec des tailles différentes de vocabulaire (10 000, 50 000 et 100 000). Malgré que la perte sur le jeu de validation avait tendance à diminuer plus la taille du vocabulaire augmentait (10 000 : perte de 149,20 ; 50 000 : perte de 84,16 ; 100

000 : perte de 58,36), la moyenne des résultats du macro-F1 calculé sur le jeu de tests (0,779) restait dans la moyenne de l'expérience précédente (0.778). Par conséquent, le caractère aléatoire de l'initialisation des poids semble avoir un grand impact sur les performances du modèle. Afin d'y remédier, il aurait été pertinent d'entraîner plusieurs fois chaque modèle avec les mêmes combinaisons d'hyperparamètres et calculer la moyenne. De plus, il aurait été intéressant de s'inspirer des travaux de Xu, Zhou, Gan, Zheng & Li (2021), qui utilisent l'entropie et la taille du vocabulaire, afin de déterminer la taille optimale du vocabulaire avant l'entraînement.

Les deux expériences précédentes ont permis de déterminer si l'utilisation des techniques d'annotation de séquences permet de généraliser l'extraction d'enregistrements et d'attributs dans une page Web contenant des sujets de discussions. Suite aux résultats, la conclusion tend vers l'affirmative. En effet, le modèle utilisé pour procéder à l'annotation de séquences (sous-objectif 1), combiné aux algorithmes d'extraction (sous-objectif 2), est en mesure d'extraire des sujets de discussions sur de nouvelles sources n'ayant pas servi durant l'entraînement (avec un résultat de 0,995 de macro F1 sur les données de test).

6.2 Expérience multisource fractionnaire

La troisième expérience réalisée avait pour but de mesurer les performances de la solution développée dans un scénario multisource fractionnaire. L'ensemble B a été utilisé afin d'entraîner un modèle avec les hyperparamètres ayant le mieux performés lors des expériences multisource classique. Lors des tests, les résultats se sont avérés extrêmement concluants. En effet, pour chacune des sources, la macro F1 calculée sur le jeu de tests s'est élevée à plus de 0,97, et ce, même pour des sources dont le nombre d'échantillons est marginal, ou dont la langue est peu commune (exemple : S15, S28).

Ces résultats tendent à démontrer que la méthode choisie est en mesure de généraliser l'extraction des sujets de discussions et de ses attributs pour de nouvelles sources, mais aussi pour les sources qui ont été utilisées lors de l'entraînement.

6.3 Erreurs générées par la solution

Lors des différentes expériences (multisource classique et fractionnaire), il était possible d'observer les erreurs de classification générées par la solution. Dans de nombreux cas, le modèle avait de la difficulté à extraire l'ensemble des dates d'une page Web. Une théorie possible pour cela est que le modèle n'a pas été en mesure de s'entraîner avec toutes les combinaisons possibles de dates existantes (jour, mois, années). Par conséquent, il aurait été intéressant de réaliser une expérience où les caractères numériques sont remplacés par un jeton générique.

6.4 Structure des pages HTML

Afin d'utiliser des techniques d'annotation de séquences sur les pages HTML, une structure nommée *graphe_HTML* a été introduite (voir section 3.1.5). L'utilisation de cette structure a permis de faciliter le nettoyage des pages HTML (voir section 3.1.6) puisqu'il était possible de naviguer au sein de la structure et retirer des banches du graphe jugées non nécessaires. De plus, cette structure à l'avantage d'associer des métadonnées à chaque noeud du *graphe_HTML*. Par exemple, il a été possible d'associer chaque noeud ou feuille du graphe à son étiquette (aucune, titre, auteur ou date) selon des règles d'extractions (voir section 3.1.2). De plus, cette structure permet de passer d'un état structuré (*graphe_HTML*) en une séquence composée de jetons représentant une balise HTML ou du texte, et repasser à un structuré. Par conséquent, il a été possible de générer une séquence partir du *graphe_HTML* représentant une page Web, annoté celle-ci, et finalement reconstruire le graphe avec la séquence étiquetée. Lors de la reconstruction, des métas-données étaient ajoutés aux noeuds et feuilles, soient les prédictions du modèle. Il était alors possible d'utiliser des algorithmes d'extractions utilisant les propriétés des graphes (voir section 3.3.3).

L'utilisation de cette structure a permis l'utilisation d'un format simplifié lors du processus d'annotation de séquences. En effet, il a été possible de limiter le nombre de jetons en utilisant seulement les balises HTML et le texte dans une séquence, tout en conservant le concept de relation parent-enfant des composants d'une page HTML (voir section 4.2). Malgré qu'il est

difficile de confirmer si ce format a contribué aux bonnes performances de la solution, il est à penser que celui-ci a aidé. En effet, les balises HTML présentent dans la séquence ont pu servir de repère au modèle, et ce, indépendamment de la langue du forum (les balises HTML sont indépendantes de la langue d'un site Web). En effet, chaque attribut étant entouré par une balise HTML, ceux-ci ont pu servir de bornes de départ et de fin.

6.5 Performance de la solution dans un contexte industriel

Cette recherche s'est faite avec un partenaire de recherche, soit l'entreprise Flare Systems. Comme mentionné dans la section 1.3, et spécifié dans la section 2.1.1, cette compagnie utilise un système qui nécessite une intervention humaine afin d'écrire et maintenir des règles d'extractions. Suite aux résultats encourageant des expériences précédentes, un système d'extraction automatique a été mis en place. Celui-ci utilise les outils développés lors de cette recherche, soit la transformation d'une page HTML en une séquence (section 3.1), l'annotation de celle-ci avec le modèle BiLSTM-CRF (section 3.2), et l'extraction des enregistrements et attributs avec les algorithmes d'extractions (section 3.3).

Suite à l'implémentions de la solution automatique, il est possible de comparer ces résultats avec le système reposant sur les règles manuelles [1]. Pour ce faire, les deux systèmes roulent en parallèle et retournent chacun les résultats pour chacune des pages Web demandées. Il est alors possible d'observer que la solution automatique fonctionne de façon presque parfaite sur les sites Web qui ont servi lors de l'entraînement. En effet, très peu d'erreurs ont été observées. Dans un seul cas, le système automatique a retourné un titre qui n'en était pas un. De plus, il est arrivé dans certains cas que la solution automatique retourne plus de résultats que le système reposant sur les règles manuelles. Cela s'explique, comme mentionné dans la section 2.1.1, par la présence de variations dans la structure interne des sites Web. Par conséquent, dans certains cas, les règles d'extraction manuelles échouent à extraire le contenu puisqu'elles ne sont pas adaptées à ces variations imprévues.

[1] Lors des observations, le modèle utilisé était celui entraîné lors de l'expérience de recherche des meilleurs hyperparamètres avec l'identifiant 1 (voir section 4.3.1)

Comme mentionné dans la section 1.2, de nouveaux sites Web émergent constamment (La compagnie Flare Systems ajoute constamment de nouveaux sites). De ce fait, il a été possible de tester le système automatique avec ceux-ci, puisqu'ils n'ont donc pas participé aux processus d'entrainement, de validation ou bien de test. Sur ces nouvelles sources, très peu d'erreurs ont encore une fois été observées. Le système est en mesure d'extraire correctement les titre, auteur et date de publication des sujets de discussions sur ces nouvelles sources. Le tableau 6.1 illustre la sortie des deux systèmes pour l'attribut «Titre» d'une page Web provenant d'une nouvelle source. La case vide dans la colonne du système automatique est due au fait que celui-ci n'a pas retourné de titre. Après observation, il s'est avéré que celui-ci avait raison, car il ne s'agissait pas d'un titre, mais bien d'un filtre présenté en tant qu'en-tête de tableau.

Fait intéressant, dans aucun cas le modèle automatique n'a retourné une date de publication pour une source dont les sujets de discussions n'en contiennent pas. De plus, le système automatique a permis de corriger des erreurs humaines. En effet, dans certains cas, l'opérateur avait indiqué que la date de la dernière personne ayant commenté était la date de publication du sujet de discussion.

Tableau 6.1 Titres prédits par les deux systèmes pour une page Web issue d'une nouvelle source

Système basé sur les règles manuelles	**Système automatique**
Eduma v4.4.2 - Education WordPress Theme	Eduma v4.4.2 - Education WordPress Theme
Autozone v5.3.6 - car dealership WordPress template	Autozone v5.3.6 - car dealership WordPress template
Jupiter X v1.24.0 - Multipurpose WordPress Theme	Jupiter X v1.24.0 - Multipurpose WordPress Theme
nulled azures	
The Landscaper v2.6.1 - a template on the theme of landscape design WordPress	The Landscaper v2.6.1 - a template on the theme of landscape design WordPress
KiviCare v1.4.1 - Medical Clinic and Patient Management WordPress Theme	KiviCare v1.4.1 - Medical Clinic and Patient Management WordPress Theme
Uniq v2.0.6 - Minimal Creative	Uniq v2.0.6 - Minimal Creative

6.6 Évolution de la solution dans un contexte industriel

Afin de faire évoluer la solution dans un contexte industriel, il est important de s'attarder aux deux grandes limitations de la solution développée, soit la nécessité d'avoir des données annotées pour entraîner le modèle et l'évolution constante du vocabulaire utilisé par les cybercriminels ainsi que de la structure des pages Web.

Dans le but d'obtenir des données annotées, il a été décidé lors de cette recherche d'utiliser des règles d'extraction afin d'annoter les pages Web contenues dans les archives de la compagnie. Cela permet d'éviter de devoir manuellement annoter des pages Web, mais cela a le désavantage de dépendre de règles d'extractions écrites par des humains pouvant comporter des erreurs (voir section 2.1.1).

Pour ce qui est de la deuxième limitation, soit l'évolution du domaine (vocabulaire, structure des pages Web), il a été convenu lors de l'élaboration de la solution de mettre une emphase sur l'automatisation du développement de celle-ci, afin de faciliter la mise à jour du modèle. Par conséquent, un pipeline a été développé permettant d'automatiser chaque étape de développement de la solution, soit :

- L'acquisition des données (pages Web) à partir des archives de l'entreprise. Celles-ci sont alors annotées avec les règles d'extractions préalablement écrites par les opérateurs humains. Les pages Web annotées sont alors transformées en *graphe_HTML* et nettoyées selon les critères énumérés à la section 3.1.6.
- L'entraînement d'un algorithme de mise en jetons de type BPE à partir des archives de l'entreprise.
- La vérification des *graphe_HTML* afin d'assurer que ceux-ci soient bien annotés selon les critères énumère à la section 3.1.4
- La mise en jeton du texte présent dans les *graphe_HTML* avec l'algorithme de mise en jeton précédemment entraîné.
- La création d'une base de données sous un format permettant l'entraînement du modèle BiLSTM-CRF.

- L'entraînement du modèle BiLSTM-CRF 3.3.5.
- L'évaluation du modèle tel que défini dans la section 3.3.5. 3.1.4.

La figure 6.1 illustre les différentes étapes du pipeline.

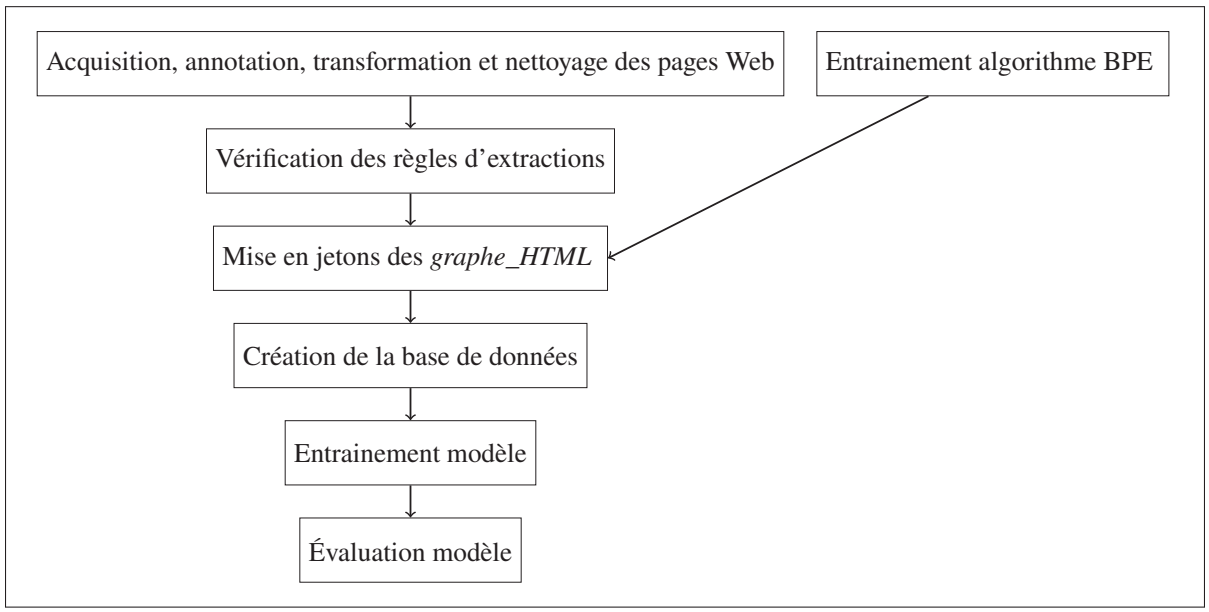


Figure 6.1 Pipeline permettant l'automatisation du développement de la solution dans un contexte industriel

Afin d'implémenter ce pipeline, de nombreux outils ont été utilisés, tels que Airflow [2] pour l'orchestration des différentes étapes, Apache Beam [3] et Flink [4] pour le traitement des données et un environnement Kubernetes [5] afin de gérer ressources nécessaires.

La mise en place de ce pipeline permet alors à l'entreprise de facilement générer un nouveau modèle si le modèle en place n'est plus en mesure d'extraire le contenu sur de nouvelles sources. De plus, des expériences préliminaires ont permis de constater qu'il est possible de se servir de l'ancien modèle afin d'annoter de nouvelles sources pour l'entraînement, en lieu et place des

[2] https ://airflow.apache.org

[3] https ://beam.apache.org

[4] https ://flink.apache.org

[5] https ://kubernetes.io

règles d'extraction (les erreurs étant attrapés par l'étape de vérification du pipeline et retirée du jeu d'entraînement).

Dans le but d'utiliser la solution pour d'autres types de pages Web, telles les pages contenant les discussions dans un forum, la solution devra être adaptée afin de tenir en compte différents types d'attributs. En effet, dans le cas des sujets de discussion, un attribut était encapsulé par une balise HTML. Il y a alors une relation un à un pour une balise HTML et l'attribut qu'elle contient. Dans le cas des pages de discussions, une discussion peut être dans plusieurs balises. Par conséquent, il faut extraire la balise parente qui contient les autres balises faisant partie de la discussion.

6.7 Contribution de cette recherche

Il est difficile de comparer ces résultats avec d'autres recherches, puisque l'utilisation des outils d'annotation de séquences est nouvelle dans le domaine de l'extraction de contenus sur le Web. Cependant, il est pertinent d'analyser la conclusion de Zhou & Mashuq (2014), qui ont tenté le même style d'expérience, soit multisource classique et fractionnaire dans un contexte d'extraction de contenu sur des pages avec un seul enregistrement (voir section 1.3) en se servant des caractéristiques visuelles d'une page Web (attributs CSS et rendu des différentes sections d'une page Web). Lors de l'expérience multisource fractionnaire, ceux-ci ont obtenu une F1 de 92.83%, comparativement à 31.28% pour l'approche classique.

Il est cependant difficile de comparer directement les résultats de cette recherche avec les leurs, dû au contexte différent (extraction du corps d'un article comparativement à l'extraction d'enregistrements et ses attributs). Cependant, leur conclusion, suggérant que les caractéristiques visuelles des sites Web sont difficilement généralisables, est intéressante à approfondir.

En effet, aucune caractéristique visuelle n'a été utilisée dans le cadre de ce mémoire. Seule la structure (balises HTML) et le texte ont été conservés afin de procéder à la classification des éléments dans une page Web. Cette approche a permis de développer une solution qui est en mesure d'être généralisée à de nouveaux sites Web, indépendamment des caractéristiques visuelles. De

ce fait, malgré que de nombreux sites Web semblent se ressembler, les caractéristiques visuelles de ceux-ci ne sont pas suffisantes pour généraliser l'extraction des éléments s'y trouvant. La structure de ceux-ci semble alors être un élément important lorsque vient le temps de généraliser l'extraction de contenu, en plus du texte encapsulé par celle-ci.

Finalement les résultats obtenus lors des précédentes expériences, ainsi que les performances de la solution dans un contexte d'exploitation commerciale, permettent d'affirmer l'hypothèse qu'il est possible de traiter une page Web comme une séquence de texte et de balises HTML. De ce fait, les contributions de ce travail de recherche sont :

- Une méthodologie permettant de transformer une page HTML en une séquence composée de balises HTML et de texte. De plus, ce mémoire apporte une solution (mise en jeton avec BPE) à la problématique des mots mal orthographiés et uniques (voir section 1.4) contenus sur les pages Web contenant du contenu généré par les utilisateurs, tels les forums.
- L'utilisation d'outils d'annotation de séquences sur une séquence issue d'une page Web.
- Des algorithmes permettant l'extraction d'enregistrements et d'attributs à partir d'une page Web annotée par annotation de séquences.
- Différentes techniques (mode stricte et souple) afin d'évaluer les performances de la solution.

CONCLUSION ET RECOMMANDATIONS

La prolifération de plateformes en ligne prônant un mode de communication anonyme, tels des forums, facilite l'échange entre acteurs malicieux sur le Web. De ce fait, surveiller ces échanges permet l'acquisition de renseignements en lien avec les futurs angles d'attaques. En effet, selon une recherche menée par Recorded Future, 75% des vulnérabilités logicielles ont été mentionnées lors d'échanges entre ces acteurs avant leur publication dans le registre officiel des vulnérabilités logiciel américain [1] (NVD) (Ladd, 2017). Par conséquent, une surveillance en temps quasi réel de ces plateformes permet aux entreprises et agences gouvernementales de comprendre les futurs angles d'attaques, et de ce fait, mieux protéger ses actifs.

Surveiller ces plateformes à des fins de prévention présente cependant de nombreux défis. Tout d'abord, cette surveillance doit se faire en continu sur une quantité en constante évolution de site Web. De plus, aucune norme n'existe sur le Web quant à la façon de présenter et structurer les données. Par conséquent, des outils provenant du domaine du moissonnage du Web doivent être mis en place afin d'extraire les données encapsulées dans des mises en page prédéfinies (HTML). Finalement, les variations terminologiques (ex : users, user$, usrs) utilisée par les communautés présentes sur ses plateformes rendent difficile toute analyse sémantique traditionnelle qui permettrait l'extraction d'éléments.

L'objectif de ce mémoire était alors de développer une méthode afin d'automatiser l'extraction des sujets de discussions dans le but de succéder aux méthodes traditionnelles s'appuyant sur des opérateurs humains qui doivent écrire et maintenir des règles d'extraction. Pour chaque sujet de discussion extrait, la solution devait aussi être en mesure d'identifier le titre, l'auteur et la date de publication de celui-ci. Afin d'y arriver, l'hypothèse qu'il est possible d'utiliser des techniques d'annotation de séquences sur une page Web a été posée. Par conséquent, deux

[1] https ://nvd.nist.gov

sous-objectifs ont été poursuivis, soit l'identification d'attributs dans une séquence composée de balises HTML et de texte, et l'extraction de ceux-ci.

Afin de répondre à ces objectifs, une structure a été mise en place afin de faciliter la transformation d'une page HTML en une séquence, et vice versa. Par conséquent, un modèle BiLSTM-CRF est utilisé afin d'identifier les attributs dans la séquence composée de texte et de balises HTML. Par la suite, cette séquence est utilisée afin de reconstruire la page Web. Des algorithmes d'extraction développés afin de répondre au deuxième sous-objectif sont alors en mesure d'extraire les enregistrements et ses attributs à partir de la page Web reconstruite.

Le résultat final est une solution qui n'exige aucune intervention humaine lors de l'utilisation, qui est en mesure d'extraire des sujets de discussions ainsi que ses attributs. Celle-ci est en mesure de procéder à cette extraction sur les sites Web qui ont servi lors de l'entraînement du modèle, mais aussi sur de nouveaux sites Web. Les bonnes performances de la solution ont pu être testées sur les données de test durant l'entraînement, mais aussi en utilisant la solution développée dans un contexte industriel.

Suite aux bons résultats de la solution développés, l'hypothèse comme quoi il est possible d'utiliser des outils d'annotation de séquences sur des pages Web se confirme. Par conséquent, il serait intéressant à l'avenir d'appliquer cette approche sur d'autres types de pages Web contenue dans les forums, telles les pages de profil d'utilisateur ou bien de discussion en lien avec un sujet (voir section 1.4). De plus, la solution pourrait être étendue à des sites Web de type place de marchés. De ce fait, il serait possible de faire des liens entre des publications sur des forums et des articles en vente en lien avec celles-ci.

ANNEXE I

DONNÉES

1. Ensembles A

Tableau-A I-1 Ensemble A

Source	Échantillon total	Langue	Structure	Attribut «Date»
Entrainement				
S2	1200	Russe	Division	X
S3	1200	Russe	Division	X
S4	1079	Russe	Division	X
S5	739	Anglais, Russe	Tableau	
S6	1200	Anglais	Tableau	X
S7	1200	Anglais	Tableau	
S8	1200	Anglais, Multiples	Liste	X
S9	1200	Anglais	Tableau	
S10	1200	Anglais	Division	X
S11	895	Anglais	Division	X
S12	1200	Anglais	Division	X
S13	1200	Russe	Division	X
S15	68	Francais	Divison	
S17	1200	Anglais, Russe	Tableau	
S19	1200	Anglais	Division	X
S20	1200	Russe	Division	X
S23	1200	Anglais	Liste	X
S24	1200	Anglais	Division	X
S25	1200	Anglais	Tableau	X
S27	948	Anglais	Liste	X
S28	198	Francais , Anglais	Liste	X
S29	1200	Anglais	Tableau	X
Validation				
S1	100	Anglais	Division	X
S16	100	Anglais	Tableau	
S22	100	Russe	Tableau	
S21	86	Anglais	Division	X
Test				
S14	100	Anglais, Russe	Liste	X
S18	100	Anglais	Tableau	
S26	100	Anglais	Division	X

1.1 Expérience 1 : Recherche des meilleurs hyperparamètres pour le modèle

Tableau-A I-2 Macro F1 sur le jeu de test de l'ensemble A lors de la recherche des meilleurs hyperparamètres

ID	DP	CB	HD	TA	DP	Perte (Validation)	Macro F1 (Test)
1	100.0	2.0	64.0	0.01	0.1	120,150	0,995
2	16.0	2.0	64.0	0.01	0.1	138,146	0,989
3	300.0	1.0	64.0	0.001	0.0	339,600	0,979
4	300.0	1.0	64.0	0.01	0.1	122,760	0,967
5	16.0	1.0	128.0	0.01	0.0	200,790	0,957
6	64.0	2.0	128.0	0.001	0.0	428,818	0,948
7	300.0	1.0	128.0	0.01	0.0	243,835	0,937
8	100.0	1.0	64.0	0.01	0.1	127,381	0,931
9	16.0	1.0	64.0	0.001	0.0	219,551	0,913
10	16.0	2.0	128.0	0.01	0.1	57,738	0,880
11	100.0	2.0	128.0	0.01	0.0	127,758	0,880
12	300.0	2.0	128.0	0.001	0.0	826,388	0,869
13	64.0	2.0	128.0	0.001	0.1	172,257	0,844
14	64.0	2.0	128.0	0.01	0.0	137,149	0,841
15	64.0	1.0	64.0	0.001	0.0	239,700	0,840
16	300.0	2.0	64.0	0.001	0.0	335,630	0,834
17	100.0	2.0	128.0	0.001	0.0	145,164	0,819
18	100.0	1.0	128.0	0.01	0.1	60,158	0,815
19	64.0	1.0	128.0	0.01	0.1	115,446	0,811
20	16.0	2.0	64.0	0.001	0.0	145,630	0,805
21	64.0	1.0	128.0	0.001	0.0	184,284	0,803
22	64.0	2.0	64.0	0.001	0.1	193,541	0,802
23	300.0	1.0	64.0	0.01	0.0	169,493	0,792

Continue à la prochaine page

Tableau-A I-2 – *Continuation de la page précédente*

ID	DP	CB	HD	TA	DP	Perte (Validation)	Macro F1 (Test)
24	100.0	1.0	64.0	0.001	0.0	658,762	0,790
25	64.0	1.0	128.0	0.01	0.0	159,575	0,775
26	300.0	1.0	128.0	0.001	0.1	120,839	0,774
27	64.0	1.0	64.0	0.01	0.0	242,721	0,773
28	16.0	1.0	128.0	0.001	0.0	159,387	0,772
29	300.0	2.0	64.0	0.001	0.1	338,603	0,765
30	300.0	2.0	128.0	0.01	0.0	208,825	0,755
31	16.0	2.0	64.0	0.01	0.0	200,550	0,753
32	16.0	2.0	128.0	0.001	0.0	379,009	0,753
33	16.0	2.0	128.0	0.01	0.0	254,539	0,752
34	100.0	1.0	128.0	0.01	0.0	220,523	0,747
35	100.0	2.0	128.0	0.01	0.1	115,394	0,746
36	100.0	1.0	64.0	0.01	0.0	144,156	0,746
37	64.0	2.0	64.0	0.01	0.1	148,985	0,744
38	100.0	1.0	128.0	0.001	0.0	307,359	0,744
39	64.0	2.0	64.0	0.001	0.0	376,561	0,744
40	100.0	2.0	128.0	0.001	0.1	450,939	0,744
41	64.0	2.0	128.0	0.01	0.1	103,529	0,743
42	100.0	2.0	64.0	0.01	0.0	140,347	0,743
43	64.0	2.0	64.0	0.01	0.0	241,179	0,743
44	100.0	2.0	64.0	0.001	0.1	669,370	0,743
45	16.0	1.0	128.0	0.01	0.1	108,274	0,742
46	16.0	2.0	64.0	0.001	0.1	192,387	0,742
47	300.0	2.0	64.0	0.01	0.0	233,257	0,742
48	64.0	1.0	64.0	0.01	0.1	79,677	0,741

Continue à la prochaine page

Tableau-A I-2 – *Continuation de la page précédente*

ID	DP	CB	HD	TA	DP	Perte (Validation)	Macro F1 (Test)
49	300.0	1.0	128.0	0.01	0.1	125,350	0,740
50	16.0	1.0	64.0	0.01	0.1	114,550	0,738
51	300.0	2.0	64.0	0.01	0.1	121,002	0,738
52	300.0	2.0	128.0	0.001	0.1	282,858	0,738
53	16.0	1.0	64.0	0.01	0.0	187,127	0,736
54	300.0	1.0	128.0	0.001	0.0	402,355	0,735
55	64.0	1.0	64.0	0.001	0.1	399,911	0,723
56	16.0	1.0	64.0	0.001	0.1	697,168	0,723
57	16.0	2.0	128.0	0.001	0.1	315,369	0,715
58	100.0	2.0	64.0	0.001	0.0	476,756	0,714
59	16.0	1.0	128.0	0.001	0.1	128,903	0,660
60	100.0	1.0	128.0	0.001	0.1	271,944	0,649
61	300.0	1.0	64.0	0.001	0.1	423,681	0,617
62	100.0	1.0	64.0	0.001	0.1	380,895	0,564
63	300.0	2.0	128.0	0.01	0.1	121,652	0,495
64	64.0	1.0	128.0	0.001	0.1	209,450	0,491

BIBLIOGRAPHIE

Abu, M., Rahayu, S., Ariffin (DrAA), D. A. & Robiah, Y. (2018). Cyber Threat Intelligence – Issue and Challenges. 10, 371–379. doi : 10.11591/ijeecs.v10.i1.pp371-379.

Aguilar, G. (2020). *Neural Sequence Labeling on Social Media Text*. (Thesis). Repéré à https://uh-ir.tdl.org/handle/10657/7726.

Arasu, A. & Garcia-Molina, H. (2003). Extracting structured data from Web pages. *Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, (SIGMOD '03), 337–348. doi : 10.1145/872757.872799.

Audeh, B., Beigbeder, M., Zimmermann, A., Jaillon, P. & Bousquet, C. (2017). Vigi4Med Scraper : A Framework for Web Forum Structured Data Extraction and Semantic Representation. 12(1), e0169658. doi : 10.1371/journal.pone.0169658. Publisher : Public Library of Science.

Bao, Z., Huang, R., Li, C. & Zhu, K. Q. (2019). Low-Resource Sequence Labeling via Unsupervised Multilingual Contextualized Representations. Repéré à http://arxiv.org/abs/1910.10893.

Baskaran, U. & Ramanujam, K. (2018). Automated scraping of structured data records from health discussion forums using semantic analysis. 10, 149–158. doi : 10.1016/j.imu.2018.01.003.

Bin Mohd Azir, M. A. & Ahmad, K. B. (2017). Wrapper approaches for web data extraction : A review. *2017 6th International Conference on Electrical Engineering and Informatics (ICEEI)*, pp. 1–6. doi : 10.1109/ICEEI.2017.8312458.

Blanc, A., Hétu, D. & Lavoie, M. (2021, Juillet, 7). Fireside Chat The Role of the Dark Web in the Canadian Companies Threat Landscape [Vidéo Youtube]. Repéré à https://www.youtube.com/watch?v=nsafzCdx9rs&t=411s.

Bronzi, M., Crescenzi, V., Merialdo, P. & Papotti, P. (2013). Extraction and integration of partially overlapping web sources. 6(10), 805–816. doi : 10.14778/2536206.2536209.

Carle, V. (2020). *Web Scraping using Machine Learning*. Repéré à http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-281344.

Chu, Y.-C., Hsu, C.-C., Lee, C.-J. & Tsai, Y.-T. (2015). Automatic data extraction of websites using data path matching and alignment. *2015 Fifth International Conference on Digital Information Processing and Communications (ICDIPC)*, pp. 60–64. doi : 10.1109/ICDIPC.2015.7323006.

Clay, E. (2021). Digital Risk Monitoring : A Quick Introduction. Repéré le 2021-10-19 à https://flare.systems/resource-center/blog/digital-risk-monitoring-a-quick-introduction/.

Cording, P. H. (2011a). Algorithms for Web Scraping. Repéré à /paper/Algorithms-for-Web-Scraping-Cording/d0f265e81bebc72024de15867388965c6b563972.

Cording, P. H. (2011b). Algorithms for Web Scraping. Repéré à /paper/Stable-web-scraping%3A-an-approach-based-on-neighbour-Gao-Han/c7e461822116a8d3173274b8b2d0c1c0cac2cef3.

CyberEdge Group, L. (2020). *2020 Cyberthreat Defense Report*. CyberEdge Group, LLC.

Dallmeier, E. C. (2021). Computer Vision-based Web Scraping for Internet Forums. *2021 7th International Conference on Optimization and Applications (ICOA)*, pp. 1–5. doi : 10.1109/ICOA51614.2021.9442634.

Den Berg, B. V. (2020). Using the Dark Web for Threat Intelligence. Repéré le 2021-10-21 à https://apmg-international.com/article/using-dark-web-threat-intelligence.

Dilmegani, C. (2020). Complete Guide to Web Scraping for Tech Buyers. Repéré le 2021-10-19 à https://research.aimultiple.com/web-scraping/.

Diouf, R., Sarr, E. N., Sall, O., Birregah, B., Bousso, M. & Mbaye, S. N. (2019). Web Scraping : State-of-the-Art and Areas of Application. *2019 IEEE International Conference on Big Data (Big Data)*, pp. 6040–6042. doi : 10.1109/BigData47090.2019.9005594.

Ferrara, E. & Baumgartner, R. (2011a). Automatic Wrapper Adaptation by Tree Edit Distance Matching. *Combinations of Intelligent Methods and Applications*, (Smart Innovation, Systems and Technologies), 41–54. doi : 10.1007/978-3-642-19618-8_3.

Ferrara, E. & Baumgartner, R. (2011b). Intelligent Self-repairable Web Wrappers. *AI*IA 2011 : Artificial Intelligence Around Man and Beyond*, (Lecture Notes in Computer Science), 274–285. doi : 10.1007/978-3-642-23954-0_26.

Ferrara, E., De Meo, P., Fiumara, G. & Baumgartner, R. (2014). Web data extraction, applications and techniques : A survey. *Knowledge-based systems*, 70, 301–323.

Forney, G. (1973). The viterbi algorithm. 61(3), 268–278. doi : 10.1109/PROC.1973.9030. Conference Name : Proceedings of the IEEE.

Gao, P., Han, H., Guo, J. & Saeki, M. (2018). Stable web scraping : an approach based on neighbour zone and path similarity of page elements. *International Journal of Web Engineering and Technology*, 13(4), 301–333.

Guccione, D. (2021). What is the dark web ? How to access it and what you'll find. Repéré le 2021-07-14 à https://www.csoonline.com/article/3249765/

what-is-the-dark-web-how-to-access-it-and-what-youll-find.html.

He, Z., Wang, Z., Wei, W., Feng, S., Mao, X. & Jiang, S. (2020). A Survey on Recent Advances in Sequence Labeling from Deep Learning Models. Repéré à http://arxiv.org/abs/2011.06727.

Heinzerling, B. & Strube, M. (2017). BPEmb : Tokenization-free Pre-trained Subword Embeddings in 275 Languages. Repéré à http://arxiv.org/abs/1710.02187.

Holt, T. J., Strumsky, D., Smirnova, O. & Kilger, M. (2012). Examining the Social Networks of Malware Writers and Hackers. 6(1), 13.

Huang, Z., Xu, W. & Yu, K. (2015). Bidirectional LSTM-CRF Models for Sequence Tagging. Repéré à http://arxiv.org/abs/1508.01991.

Hétu, D. (2021). 5 Dark Web Questions Security Professionals Need Answered. Repéré le 2021-07-15 à https://flare.systems/resource-center/blog/5-dark-web-questions-security-professionals-need-answered/.

jidasheng. (2019). jidasheng/bi-lstm-crf. Repéré à https://github.com/jidasheng/bi-lstm-crf.

Kadoguchi, M., Hayashi, S., Hashimoto, M. & Otsuka, A. (2019). Exploring the Dark Web for Cyber Threat Intelligence using Machine Leaning. *2019 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pp. 200–202. doi : 10.1109/ISI.2019.8823360.

Kushmerick, N. (2000). Wrapper induction : Efficiency and expressiveness. *Artificial intelligence*, 118(1-2), 15–68.

Ladd, B. (2017). The Race Between Security Professionals and Adversaries. Repéré le 2021-10-24 à https://www.recordedfuture.com/vulnerability-disclosure-delay/.

Lafferty, J., McCallum, A. & Pereira, F. C. (2001). Conditional random fields : Probabilistic models for segmenting and labeling sequence data.

Li, J. & Ezeife, C. I. (2006). Cleaning Web Pages for Effective Web Content Mining. *Database and Expert Systems Applications*, (Lecture Notes in Computer Science), 560–571. doi : 10.1007/11827405_55.

Liu, W., Yan, H. & Xiao, J. (2011). Automatically extracting user reviews from forum sites. *Computers & Mathematics with Applications*, 62(7), 2779–2792.

Macdonald, M., Frank, R., Mei, J. & Monk, B. (2015). Identifying Digital Threats in a Hacker Web Forum. *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, (ASONAM '15), 926–933.

doi : 10.1145/2808797.2808878.

Manica, E., Dorneles, C. F. & Galante, R. (2017). R-Extractor : A Method for Data Extraction from Template-Based Entity-Pages. *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, 1, 778–787. doi : 10.1109/COMPSAC.2017.202.

Mohurle, S. & Patil, M. (2017). A brief study of wannacry threat : Ransomware attack 2017. *International Journal of Advanced Research in Computer Science*, 8(5), 1938–1940.

Morgan, S. (2017). Is cybercrime the greatest threat to every company in the world ? Repéré le 2021-07-14 à https://www.csoonline.com/article/3210912/is-cybercrime-the-greatest-threat-to-every-company-in-the-world.html.

Morgan, S. (2019). *2019 Official Annual Cybercrime Report*. Herjavec.

Morgan, S. [Section : Reports]. (2020). Cybercrime To Cost The World $10.5 Trillion Annually By 2025. Repéré le 2021-10-14 à https://cybersecurityventures.com/cybercrime-damage-costs-10-trillion-by-2025/.

Nasraoui, O. (2008). Web data mining : Exploring hyperlinks, contents, and usage data. *ACM SIGKDD Explorations Newsletter*, 10(2), 23–25.

Ng, C. H., Ng, C. J. & Lim, T. M. (2019). VFX : A VISION-BASED APPROACH TO FORUM DATA EXTRACTION. 8.

Nunes, E., Diab, A., Gunn, A., Marin, E., Mishra, V., Paliath, V., Robertson, J., Shakarian, J., Thart, A. & Shakarian, P. (2016). Darknet and deepnet mining for proactive cybersecurity threat intelligence. *2016 IEEE Conference on Intelligence and Security Informatics (ISI)*, pp. 7–12. doi : 10.1109/ISI.2016.7745435.

Pastrana, S., Thomas, D. R., Hutchings, A. & Clayton, R. (2018). CrimeBB : Enabling Cybercrime Research on Underground Forums at Scale. *Proceedings of the 2018 World Wide Web Conference*, (WWW '18), 1845–1854. doi : 10.1145/3178876.3186178.

Portnoff, R. S., Afroz, S., Durrett, G., Kummerfeld, J. K., Berg-Kirkpatrick, T., McCoy, D., Levchenko, K. & Paxson, V. (2017a). Tools for Automated Analysis of Cybercriminal Markets. *Proceedings of the 26th International Conference on World Wide Web*, pp. 657–666. doi : 10.1145/3038912.3052600.

Portnoff, R. S., Afroz, S., Durrett, G., Kummerfeld, J. K., Berg-Kirkpatrick, T., McCoy, D., Levchenko, K. & Paxson, V. (2017b). Tools for Automated Analysis of Cybercriminal Markets. *Proceedings of the 26th International Conference on World Wide Web*, (WWW '17), 657–666. doi : 10.1145/3038912.3052600.

Quattoni, A., Wang, S., Morency, L.-P., Collins, M. & Darrell, T. (2007). Hidden conditional random fields. *IEEE transactions on pattern analysis and machine intelligence*, 29(10), 1848–1852.

Sennrich, R., Haddow, B. & Birch, A. (2016). Neural Machine Translation of Rare Words with Subword Units. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pp. 1715–1725. doi : 10.18653/v1/P16-1162.

Singrodia, V., Mitra, A. & Paul, S. (2019). A Review on Web Scrapping and its Applications. *2019 International Conference on Computer Communication and Informatics (ICCCI)*, pp. 1–6. doi : 10.1109/ICCCI.2019.8821809.

Spielman, B. (2016). Dark Web intelligence automation is key for effective security [Format]. Repéré le 2021-08-04 à https://blog.cybersixgill.com/dark-web-intelligence-automation.

Ujwal, B., Gaind, B., Kundu, A., Holla, A. & Rungta, M. (2017). Classification-Based Adaptive Web Scraper. *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 125–132. doi : 10.1109/ICMLA.2017.0-168.

Venkata, S. (2005). Web Document Classification Using Edit Distances Between XML Document & Schemata. Repéré à https://digitalcommons.wku.edu/theses/3445.

VKCOM. [original-date : 2019-06-06T11 :38 :28Z]. (2021). VKCOM/YouTokenToMe. VK.com. Repéré le 2021-06-11 à https://github.com/VKCOM/YouTokenToMe.

Wallach, H. M. (2004). Conditional random fields : An introduction. *Technical Reports (CIS)*, 22.

Wang, Y., Wang, H., Zhang, L., Wang, Y., Li, J. & Gao, H. (2016). Extend tree edit distance for effective object identification. 46(3), 629–656. doi : 10.1007/s10115-014-0816-1.

Wicaksono, A. F. & Myaeng, S.-H. (2013). Automatic extraction of advice-revealing sentences foradvice mining from online forums. *Proceedings of the seventh international conference on Knowledge capture*, (K-CAP '13), 97–104. doi : 10.1145/2479832.2479857.

Xu, J., Zhou, H., Gan, C., Zheng, Z. & Li, L. (2021). Vocabulary learning via optimal transport for neural machine translation. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, pp. 7361–7373.

Zhang, J., Jin, C., Lin, Y. & Gong, X. (2012). Forum Data Extraction without Explicit Rules. *2012 Second International Conference on Cloud and Green Computing*, pp. 460–465. doi : 10.1109/CGC.2012.72.

Zhou, Z. & Mashuq, M. (2014). Web Content Extraction Through Machine Learning. 1–5.