



HAL
open science

Recherche d'entités nommées complexes sur le web : propositions pour l'extraction et pour le calcul de similarité

Armel Fotsoh Tawaofaing

► **To cite this version:**

Armel Fotsoh Tawaofaing. Recherche d'entités nommées complexes sur le web : propositions pour l'extraction et pour le calcul de similarité. Informatique. Université de Pau et des Pays de l'Adour, 2018. Français. NNT : 2018PAUU3003 . tel-04928524

HAL Id: tel-04928524

<https://theses.hal.science/tel-04928524v1>

Submitted on 4 Feb 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Recherche d'entités nommées complexes sur le Web - propositions pour l'extraction et pour le calcul de similarité

THÈSE

présentée et soutenue publiquement le 27 février 2018

pour l'obtention du

Doctorat de l'Université de Pau et des Pays de l'Adour

par

Armél FOTSOH TAWOFAING

Composition du jury

Président : Michael MARISSA, Pr., LIUPPA - Université de Pau

Rapporteurs : Sylvain PEYRONNET, Dr. HDR, IX-LABS - QWANT, Rouen
Thierry CHARNOIS, Pr., LIPN - Université Paris 13
Antoine DOUCET, Pr., L3I - Université de La Rochelle

Examineurs : Yudith CARDINALE, Pr., Université Simón Bolívar de Caracas
Tanguy MOAL, Ing., COGNITEEV, Bordeaux

Directeurs : Christian SALLABERRY, Mc. HDR, LIUPPA - Université de Pau
Annig LE PARC - LACAYRELLE, Mc., LIUPPA - Université de Pau

Mis en page avec la classe thesul.

Remerciements

Je suis très heureux d'être arrivé au terme de cette belle expérience qu'est la thèse, dure, mais très enrichissante. Je voudrais remercier toutes ces personnes qui m'ont accompagnées, de près ou de loin dans la réalisation de ce travail.

Je remercie tout d'abord Sylvain Peyronnet, Antoine Doucet et Thierry Charnois d'avoir accepté de rapporter mon mémoire. Je remercie également Yudith Cardinale, Michael Mrissa et Tanguy Moal d'être mes examinateurs.

Un grand merci à Christian Sallaberry et Annig Le Parc-Lacayrelle, mes encadrants tout au long de la thèse, merci pour la disponibilité, les conseils, les remarques, la confiance et surtout la grande rigueur dont vous avez fait montre. Cela m'a beaucoup aidé d'un point de vue personnel, et je suis ravi et très honoré d'avoir évolué sous vos ailes pendant ce projet de thèse.

Je voudrais également saisir cette occasion pour dire ma gratitude au Conseil Régional d'Aquitaine et à l'entreprise Cogniteev pour avoir financé ma thèse. Un merci particulier à Tanguy pour toutes les technologies que j'ai eues à apprendre à ses côtés, je suis devenu un « geek » tonton. Une mention spéciale à la Cogniteam d'hier (Damien, Thony, Tristan, etc.) et d'aujourd'hui (François, Lionel, Éloïse, Adrien, Emma, Anthony, Philippe, Elodie, Erle, etc.), la bonne ambiance et la compétence est votre grande force et je ne doute point : « future is yours ».

Je tiens également à remercier Albert Royer, le « vétéran », qui m'a été d'une grande aide dans la rédaction de ce mémoire, merci pour les nombreuses relectures, merci pour les nombreuses leçons de la langue de Molière, merci aussi pour les conseils et surtout pour le regard extérieur que tu m'as apporté. Je voudrais également dire merci à Mauro Gaio pour les apartés qu'il m'a accordés pour partager avec moi son expérience, m'écouter et me faire des suggestions très intéressantes.

Un merci à tous les membres du laboratoire et département informatique de l'Université de Pau et des Pays de l'Adour, particulièrement Marylène et Régine, nos secrétaires. Mention exceptionnelle à « l'État-major du midi », je veux penser à Christophe, à Mamour, à Congduc, à Ehsan, à Laurent, à Awa, à Franck, les deux Eric, à Mathieu, etc. Nos discussions m'ont constamment aidé à relâcher la pression. Vous aurez chacun votre récompense quand l'opération « conquête de l'empire » sera terminée. Merci aussi à Marie-No pour le partage des résultats de sa veille, plus d'une fois cela m'a inspiré.

Je remercie également tous les doctorants du laboratoire et particulièrement ceux de Pau (Meriem, Khaled et Amine) pour les échanges et les encouragements mutuels. Merci également à mes nombreux amis qui m'ont encouragé, accompagné et parfois conseillé durant la thèse. Merci à Eddy qui m'a aidé dans le développement du prototype, merci également à Junior et Ulrich pour les remarques et suggestions.

Je souhaite remercier particulièrement mon grand frère, le « républicain ». « Gars », merci mille fois pour mes uniques repas quotidiens que tu m'offrais gracieusement pendant la fatidique épreuve de la rédaction du mémoire. Merci à ma famille pour le soutien et surtout pour avoir supporté mes « disparitions ponctuelles », c'était pour la bonne cause. Merci à mes parents, pour les sacrifices consentis depuis toujours, merci à mes sœurs pour le soutien spirituel et émotionnel.

Ils sont nombreux ceux que je n'ai pas cité ici, mais que chacun trouve dans ce mémoire une profonde pensée pour lui et l'expression de ma gratitude.

Je voudrais finir par dire toute ma reconnaissance à Dieu, pour la santé, la forme et surtout la force qu'il m'a accordé durant ces années. Sans ces éléments, je n'aurai sûrement pas pu arriver au bout de ce long et tumultueux périple. À lui soit la gloire!

Résumé

Les récents développements des nouvelles technologies de l'information et de la communication font du Web une véritable mine d'information. Cependant, les pages Web sont très peu structurées. Par conséquent, il est difficile pour une machine de les traiter automatiquement pour en extraire des informations pertinentes pour une tâche ciblée. C'est pourquoi les travaux de recherche s'inscrivant dans la thématique de l'Extraction d'Information dans les pages web sont en forte croissance. Une fois extraites, ces informations sont généralement structurées et stockées dans des index. L'interrogation de ces index, pour répondre à des besoins d'information précis, correspond à la Recherche d'Information (RI). Notre travail de thèse se situe à la croisée de ces deux thématiques. Notre objectif principal est de concevoir et de mettre en œuvre des stratégies permettant de scruter le web pour en extraire des Entités Nommées (EN) complexes (EN composées de plusieurs propriétés pouvant être du texte ou d'autres EN) de type entreprise ou de type événement, par exemple. Nous proposons ensuite des services d'indexation et d'interrogation pour répondre à des besoins d'informations.

Ces travaux ont été réalisés au sein de l'équipe T2I du LIUPPA, et font suite à une commande de l'entreprise Cogniteev, dont le cœur de métier est centré sur l'analyse du contenu du Web. Les problématiques visées sont, d'une part, l'extraction d'EN complexes sur le Web et, d'autre part, l'indexation et la recherche d'information intégrant ces EN complexes.

Notre première contribution porte sur l'extraction d'EN complexes dans des textes. Pour cette contribution, nous prenons en compte plusieurs problèmes, notamment le contexte bruité caractérisant certaines propriétés (pour un événement par exemple, la page web correspondante peut contenir deux dates : la date de l'événement et celle de mise en vente des billets). Pour ce problème en particulier, nous introduisons un module de détection de blocs qui permet de focaliser l'extraction des propriétés sur les blocs de texte pertinents. Nos expérimentations montrent une nette amélioration du processus d'extraction dans un contexte bruité grâce à cette approche. Nous nous sommes également intéressés à l'extraction des adresses, où la principale difficulté découle du fait qu'aucun standard ne se soit réellement imposé comme modèle de référence. Nous proposons un modèle étendu et une approche d'extraction par patrons.

Notre deuxième contribution porte sur le calcul de similarité entre EN complexes. Dans l'état de l'art, ce calcul se fait généralement en deux étapes : (i) une première calcule les similarités entre propriétés et, (ii) une deuxième agrège les scores obtenus pour le calcul de la similarité globale. En ce qui concerne cette première étape, nous complétons l'état de l'art en proposant une fonction de calcul de similarité entre EN spatiales, l'une représentée par un point et l'autre par un polygone. Notons que nos principales propositions se situent au niveau de la deuxième étape. Ainsi, nous proposons trois techniques pour l'agrégation des scores intermédiaires. Les deux premières sont basées sur la somme pondérée des scores intermédiaires (combinaison linéaire et régression logistique). La troisième exploite les arbres de décisions pour agréger les scores intermédiaires. Enfin, nous proposons une dernière approche basée sur le clustering et le modèle vectoriel de Salton pour le calcul de similarité entre EN complexes. Son originalité vient du fait qu'elle ne nécessite pas de passer par le calcul de scores de similarités intermédiaires.

Mots-clés: EN Complexe, Extraction d'EN, Calcul de similarité, Recherche d'information, Web mining

Abstract

Recent developments in information technologies have made the web an important data source. However, the web content is rather unstructured. Therefore, automatically processing the web content in order to extract relevant information is a difficult task. This is a reason why research works related to Information Extraction (IE) on the web are growing quickly. Once extracted, this information is structured and stored in indexes. The parsing of indexes to answer an information need corresponds to Information Retrieval (IR). Our research work is at the crossroads of both areas. The main goal of our work is to develop strategies and techniques for crawling the web in order to extract complex Named Entities (NEs) (NEs with several properties that may be text or other NEs). We then propose to index them and to query them in order to answer information needs.

This work was carried out within the T2I team of the LIUPPA laboratory, in collaboration with Cogniteev, a company which core business is focused on the analysis of web content. The issues we had to deal with were the extraction of complex NEs on the web and the development of IR services supplied by the extracted data.

Our first contribution is related to complex NEs extraction from text content. We take into consideration several problems, in particular the noisy context characterizing some properties (a web page describing an event for example, may contain more than one date : the event's date and the date of ticket's sales opening). For this particular problem, we introduce a block detection module that focuses property's extraction on relevant text blocks. Our experiments show an improvement of our extraction system's performances in a noisy context. We also focused on address extraction where the main issue arises from the fact that there is not a standard way for writing addresses on the web. We therefore propose a pattern-based approach which uses some lexicons for extracting addresses from text, regardless of proprietary resources.

Our second contribution deals with similarity computation between complex NEs. In the state of the art, this similarity computation is generally performed in two steps : (i) first, similarities between properties are calculated ; (ii) then these similarities are aggregated to compute the overall similarity. With regard to the first step, we extend the state of the art by proposing a similarity computation function between spatial NEs, one represent by a point and the other by a polygon. However, our main proposals focuses on the second step. We propose three techniques for aggregating property similarity scores. The first two are based on the weighted sum of these scores (linear combination and logistic regression). The third technique, uses decision trees for the aggregation. Finally, we also propose a fourth approach based on clustering and Salton vector model. This last approach evaluates the similarity at the complex NE level without computing property similarity scores.

Keywords: Complex NEs, NE Extraction, Similarity Computation, Information Retrieval, Web mining

Sommaire

Table des figures	xi
Liste des tableaux	xiii
Listings	xv

I Introduction Générale 1

Chapitre 1

Le projet *Cognisearch* : une architecture de services de recherche d'entités nommées
« *entreprise* » et « *événement* »

1.1 Collaboration LIUPPA & COGNITEEV	3
1.2 Contexte de la thèse	4
1.3 Problématique	5
1.4 Contributions	9
1.5 Organisation du mémoire	9

II État de l'art : de l'extraction à la recherche d'entités nommées 11

Chapitre 2

Contexte et problèmes de recherche

Chapitre 3

Entité Nommée : définition et catégories

Chapitre 4

Modèles de représentation d'entités nommées

4.1 Catégories de modèles de représentation d'EN	19
4.1.1 Les modèles issus des standards du web	19
4.1.2 Les modèles de type ontologique	20
4.1.3 Les modèles « ad-hoc »	21

4.2	Exemples de modèles de représentation d'EN	21
4.2.1	Modèles de représentation des EN temporelles	21
4.2.2	Modèles de représentation des EN spatiales	21
4.2.3	Modèles de représentation des EN sociales	24
4.2.4	Modèles de représentation d'EN <i>entreprise</i>	26
4.2.5	Modèles de représentation d'EN <i>événement</i>	27
4.3	Bilan	31

Chapitre 5

Extraction d'entités nommées sur le Web

5.1	Du web au texte	36
5.1.1	Filtrage du web	36
5.1.1.1	Approche « focused crawling »	36
5.1.1.2	Approche « semantic crawling »	36
5.1.1.3	Approche « semantic web crawling »	37
5.1.2	Classification de pages Web	37
5.1.2.1	Approches exploitant uniquement le texte des pages	37
5.1.2.2	Approches exploitant texte, structure et médias des pages	37
5.1.3	Identification des blocs structurels d'une page	38
5.1.3.1	Approches d'identification de blocs structurels basées sur des heuristiques	38
5.1.3.2	Approches d'identification de blocs structurels basées sur de l'apprentissage	39
5.2	Approches d'extraction d'entités nommées dans le texte	39
5.2.1	Approches par patrons	40
5.2.2	Approches par apprentissage	40
5.2.3	Approches basées sur les ressources externes	41
5.2.4	Approches hybrides	42
5.3	Travaux dédiés à l'extraction d'EN	42
5.3.1	Extraction des EN temporelles	42
5.3.2	Extraction des EN spatiales	43
5.3.3	Extraction des EN sociales	44
5.3.4	Extraction d'EN <i>entreprise</i>	44
5.3.5	Extraction d'EN <i>événement</i>	44
5.3.5.1	Extraction d'EN <i>événement</i> dans les textes courts	44
5.3.5.2	Extraction d'EN <i>événement</i> dans les textes longs	45
5.4	Bilan	46

Chapitre 6

Calcul de similarité entre entités nommées

6.1	Approches de calcul de similarité entre EN	49
6.1.1	Calcul de similarité basé sur la représentation textuelle des EN	50

6.1.1.1	Approches basées sur les caractères	50
6.1.1.2	Approches basées sur les mots	50
6.1.1.3	Approches hybrides	51
6.1.2	Calcul de similarité basé sur la représentation sémantique des EN	51
6.1.2.1	Calcul de similarité basé sur le chemin le plus court	51
6.1.2.2	Calcul de similarité basé sur la profondeur	51
6.1.2.3	Calcul de similarité basé sur la fréquence d'occurrence dans un corpus	52
6.1.3	Calcul de similarité basé sur la représentation numérique des EN	52
6.1.3.1	Représentation sur un axe	52
6.1.3.2	Représentation sur un plan	53
6.1.3.3	Représentation dans l'espace	54
6.1.4	Calcul de similarité entre ensembles d'EN	54
6.1.5	Calcul de similarité entre EN complexes	55
6.2	Travaux dédiées au calcul de similarité entre EN	56
6.2.1	Calcul de similarité entre EN temporelles	56
6.2.2	Calcul de similarité entre EN spatiales	57
6.2.3	Calcul de similarité entre EN sociales	57
6.2.4	Calcul de similarité entre EN de type POI	58
6.2.5	Calcul de similarité entre EN événement	58
6.3	Bilan	59

Bilan de l'état de l'art

III Contributions à l'extraction et au calcul de similarité entre entités nommées complexes 65

Chapitre 7

Architecture <i>Cognisearch</i>
--

Chapitre 8

Modèles de représentation d'entités nommées complexes
--

8.1	Modèle de représentation d'EN <i>adresse</i>	69
8.2	Modèle de représentation d'EN <i>entreprise</i>	72
8.3	Modèle de représentation d'EN <i>événement</i>	73
8.4	Conclusion	75

Chapitre 9

Extraction d'entités nommées complexes dans un texte

9.1	Rappel du problème et contexte	77
9.2	Proposition d'une chaîne générique d'extraction d'EN complexes dans un texte	79
9.3	Extraction d'EN <i>adresse</i>	82

9.3.1	Contexte d'extraction d'EN <i>adresse</i>	82
9.3.2	Mise en œuvre de la chaîne générique pour l'extraction d'EN <i>adresse</i>	83
9.3.3	Expérimentation de l'extraction d'EN <i>adresse</i>	88
9.3.3.1	Protocole d'évaluation de l'annotation d'EN <i>adresse</i>	88
9.3.3.2	Analyse des résultats de l'annotation d'EN <i>adresse</i>	88
9.4	Extraction d'EN <i>entreprise</i>	89
9.4.1	Contexte d'extraction d'EN <i>entreprise</i>	89
9.4.2	Mise en œuvre de la chaîne générique pour l'extraction d'EN <i>entreprise</i>	89
9.4.3	Expérimentation de l'extraction d'EN <i>entreprise</i>	91
9.4.3.1	Protocole d'évaluation de l'annotation des propriétés d'EN <i>entreprise</i>	91
9.4.3.2	Analyse des résultats de l'annotation des propriétés d'EN <i>entreprise</i>	92
9.5	Extraction d'EN événement	92
9.5.1	Contexte d'extraction d'EN <i>événement</i>	93
9.5.2	Mise en œuvre de la chaîne générique pour l'extraction d'EN <i>événement</i>	93
9.5.3	Expérimentation de l'extraction d'EN <i>événement</i>	96
9.5.3.1	Protocole d'évaluation de l'annotation des propriétés d'EN <i>événement</i>	97
9.5.3.2	Analyse de résultats de l'annotation de blocs et de propriétés	97
9.6	Bilan	100

Chapitre 10

Calcul de similarité entre entités nommées complexes

10.1	Rappel du contexte et positionnement du travail	101
10.2	Proposition de quatre approches de calcul de similarité entre EN complexes	103
10.2.1	Calcul de similarité entre propriétés	104
10.2.1.1	Calcul de similarité entre propriétés de type simple	104
10.2.1.2	Calcul de similarité entre propriétés de type ensemble	106
10.2.2	Calcul de similarité entre EN complexes	107
10.2.2.1	Approche par combinaison linéaire avec pondération par étalonnage	108
10.2.2.2	Approche par régression logistique	108
10.2.2.3	Approche basée sur les arbres de décision	109
10.2.2.4	Approche basée sur le clustering et le modèle vectoriel	111
10.3	Application au calcul de similarité entre événements et entre représentations	113
10.3.1	Formalisation du problème	114
10.3.2	Protocole d'expérimentation du calcul de similarité	114
10.3.2.1	Fonctions de calcul de similarité selon le type des propriétés	114
10.3.2.2	Contexte d'expérimentation : hypothèses de travail	115
10.3.2.3	Métriques d'évaluation	116
10.3.3	Expérimentations des approches de calcul de similarité	118
10.3.3.1	Paramétrage des approches pour l'expérimentation	118
10.3.3.2	Expérimentation du calcul de la similarité entre événements	118
10.3.3.3	Expérimentation du calcul de la similarité entre représentations	124
10.4	Bilan	129

Chapitre 11**Mise en œuvre de l'architecture *Cognisearch***

11.1 Cognisearch Business	131
11.1.1 Filtrage du web	132
11.1.2 Enrichissement des ressources	135
11.1.3 Extraction d'information	136
11.1.4 Indexation	136
11.1.5 Recherche d'information	137
11.1.6 Prototype du service <i>Cognisearch Business</i>	138
11.2 Cognisearch Event	139
11.2.1 Prétraitements	140
11.2.2 Extraction d'information	144
11.2.3 Indexation	145
11.2.4 Recherche d'information	146
11.2.5 Prototype du service <i>Cognisearch Event</i>	148
11.3 Conclusion	149

IV Conclusion générale**151****Chapitre 12****Bilan : une architecture et des services génériques de recherche d'entités nommées complexes****Bibliographie**

Table des figures

1.1	Résultats de la requête « charpente traditionnelle à Ustaritz » sur les Pages Jaunes	5
1.2	Page d'accueil du site web d'une entreprise spécialisée dans la « charpente traditionnelle » à Ustaritz »	6
1.3	Événements similaires sur deux plateformes distinctes	7
2.1	Vue simplifiée du projet <i>Cognisearch</i>	13
2.2	La solution envisagée pour le projet <i>Cognisearch</i>	14
4.1	Tournée d'Eric Clapton selon l'ontologie EVENT	28
4.2	Tournée d'Eric Clapton selon l'ontologie LODÉ	28
4.3	Tournée d'Eric Clapton selon l'ontologie SEM	29
5.1	Découpage en blocs structurels d'une page web, extraite de la thèse de Faessel [51]	38
6.1	Calcul de la similarité entre EN représentées sur un axe	53
6.2	Calcul de la similarité entre EN représentées sur un plan	53
6.3	Calcul de la similarité entre EN représentées par des points de l'espace	54
7.1	Architecture <i>Cognisearch</i>	67
8.1	Notre Modèle de représentation d'EN <i>adresse</i>	70
8.2	Notre modèle de représentation d'une EN <i>entreprise</i>	72
8.3	Notre modèle de représentation d'une EN <i>événement</i>	74
9.1	Texte extrait d'une page décrivant un événement	78
9.2	Texte extrait de la page d'un site web d'entreprise	79
9.3	Chaîne générique d'extraction d'EN complexes dans un texte	80
9.4	Exemple de texte, décrivant un événement et contenant des propriétés bruitées	80
9.5	Exemple de texte, décrivant un événement avec les propriétés annotées	81
9.6	Exemple d'instanciation du patron (4) pour annoter un CA en début d'adresse	85
9.7	Exemple d'instanciation du patron (4) pour annoter un CA en milieu d'adresse	86
9.8	Instanciation du patron (6) pour annoter un nom de voie	87
9.9	Instanciation du patron (6) pour annoter un nom de voie se terminant par un nom de commune	87
9.10	Annotation de propriétés d'EN <i>entreprise</i>	90

9.11	Détection de blocs dans le texte d'un événement contenant plusieurs représentations . . .	93
9.12	Annotation des propriétés d'événements	94
10.1	Exemple d'événement référencé sur deux sites différents	102
10.2	Exemple d'un événement avec deux représentations décrites sur deux pages distinctes d'un même site web	102
10.3	Rectangle englobant un polygone	106
10.4	Arbre de décision pour le calcul de similarité	110
10.5	Partitionnement en clusters d'un ensemble de 4 EN <i>événement</i>	112
10.6	Variations entre scores de similarité de l'expert et ceux de CombMNZ	120
10.7	Variations entre scores de similarité de l'expert et ceux calculés par l'approche par étalonnage	120
10.8	Variations entre scores de similarité de l'expert et ceux calculés par régression logistique .	120
10.9	Variations entre scores de similarité de l'expert et ceux calculés en exploitant les arbres de décision	121
10.10	Variations entre scores de similarité de l'expert et ceux calculés en combinant clustering et modèle vectoriel	121
10.11	Variations entre scores de similarité de l'expert et ceux de CombMNZ	125
10.12	Variations entre scores de similarité de l'expert et ceux calculés par l'approche par étalonnage	125
10.13	Variations entre scores de similarité de l'expert et ceux calculés par régression logistique .	126
10.14	Variations entre scores de similarité de l'expert et ceux calculés en exploitant les arbres de décision	126
10.15	Variations entre scores de similarité de l'expert et ceux calculés en combinant clustering et modèle vectoriel	126
11.1	Chaîne de traitement d'EN <i>entreprise</i>	132
11.2	Filtrage du web	133
11.3	Enrichissement lexical de la ressource des activités	135
11.4	Indexation d'EN <i>entreprise</i>	137
11.5	Recherche d'information d'entreprises	138
11.6	Architecture technique du prototype du service Cognisearch Business	139
11.7	Chaîne de traitement d'EN <i>événement</i>	140
11.8	Prétraitements des pages web	141
11.9	Exemple de page « détail »	141
11.10	Exemple de page « autre »	142
11.11	Indexation d'EN <i>événement</i>	146
11.12	Interface de recherche d'événements	147
11.13	Exemple de résultat pour une requête	147
11.14	Architecture technique du prototype du service Cognisearch Event	148

Liste des tableaux

4.1	Propriétés de l'adresse de Cogniteev avec le modèle Adresse de l'Etalab	24
4.2	Propriétés principales d'une entreprise selon le modèle SIRENE	27
4.3	Quelques modèles de représentation des EN	33
5.1	Synthèse des travaux relatifs à l'extraction de quelques types d'EN	47
6.1	Synthèse des travaux relatifs au calcul de similarité entre EN élémentaires	61
6.2	Synthèse des travaux relatifs au calcul de similarité entre EN complexes	62
8.1	Propriétés d'une adresse selon notre modèle	71
8.2	Exemple d'adresse selon notre modèle	71
9.1	Règles et symboles de notations pour les patrons	84
9.2	Notations des propriétés d'adresse	84
9.3	Résultat de l'évaluation de l'annotation d'adresses	88
9.4	Évaluation de l'annotation des propriétés d'EN <i>entreprise</i>	92
9.5	Évaluation du module de détection de blocs	97
9.6	Évaluation du module d'annotation de propriétés sans détection préalable de blocs	98
9.7	Évaluation du module d'annotation de propriétés avec détection préalable de blocs	98
10.1	Jeu d'évaluation du calcul de similarité entre événements	119
10.2	Seuils optimaux pour la similarité entre événements	121
10.3	Résultats obtenus avec les différentes approches sur l'ensemble des couples d'événements du jeu d'évaluation	122
10.4	Résultats obtenus avec les différentes approches sur les couples d'événements des différentes familles	123
10.5	Résultats obtenus avec les différentes approches sur les couples d'événements partiellement renseignés	124
10.6	Jeu d'évaluation du calcul de similarité entre représentations	124
10.7	Seuils optimaux pour la similarité entre représentations	127
10.8	Résultats obtenus avec les différentes approches sur l'ensemble des couples de représentations du jeu d'évaluation	127
10.9	Résultats obtenus avec les différentes approches sur les couples de représentations des différentes familles	128

10.10	Résultats obtenus avec les différentes approches sur les couples de représentations partiellement renseignés	129
10.11	Résultats des expérimentations de Nikolov et al. [131] et Khrouf et al. [89]	130
11.1	Évaluation du module de filtrage de la chaîne <i>Cognisearch Business</i>	134
11.2	Résultats de la validation croisée	144

Listings

4.1	Représentation d'un film suivant le formalisme schema.org	20
4.2	Représentation d'EN temporelles en TimeML	21
4.3	Représentation du « Bataclan » suivant schema.org/Place	22
4.4	Représentation du « Bataclan » suivant h-adr	22
4.5	Représentation du « Bataclan » suivant GeoRSS	23
4.6	Représentation d'Eric Clapton selon le modèle h-card	24
4.7	Représentation l'entreprise Cogniteev avec la classe schema.org/Organization	25
4.8	Représentation l'entreprise Cogniteev avec la classe foaf:Organization	25
4.9	Représentation l'entreprise Cogniteev avec la classe schema.org/LocalBusiness	26
4.10	Représentation d'une prestation de la tournée d'Eric Clapton avec h-event	29
4.11	Représentation de la tournée d'Eric Clapton avec schema.org/Event	30
9.1	Exemple d'adresse suivant la norme AFNOR NF Z 10-001	82
9.2	Exemple d'adresse ne respectant pas la norme AFNOR NF Z 10-001	83

Première partie

Introduction Générale

Chapitre 1

Le projet *Cognisearch* : une architecture de services de recherche d'entités nommées « *entreprise* » et « *événement* »

Les travaux présentés dans ce mémoire de thèse ont été financés pour moitié par le *Conseil Régional de la Nouvelle-Aquitaine*¹ et l'autre moitié par l'entreprise *Cogniteev*². Ils ont été réalisés au sein du *Laboratoire Informatique de l'Université de Pau et des Pays de l'Adour (LIUPPA)*³, et plus précisément dans l'équipe *Traitement des Interactions et des Informations (T2I)*. Le cœur de métier de l'entreprise *Cogniteev* est centré sur l'analyse du contenu du web. Elle a développé à cet effet des produits qui permettent à ses clients de réaliser : (i) l'analyse structurelle de leurs sites web, en vue d'un référencement naturel efficace ; (ii) ainsi que l'analyse sémantique de ces sites afin de faciliter leur compréhension par des robots. Les attentes de *Cogniteev* auxquelles nous devons répondre visent, d'une part, l'extraction d'entités *entreprise* et *événement* du web et, d'autre part, la conception de services de recherche d'information qui exploitent ces entités. Ce chapitre présente le contexte de la thèse, nos objectifs, problématiques et contributions, avant de détailler l'organisation du manuscrit.

1.1 Collaboration LIUPPA & COGNITEEV

L'entreprise *Cogniteev* a mis au point des solutions informatiques ayant pour vocation principale d'analyser le contenu du web. De ce fait, *Cogniteev* a développé un pôle de compétences autour de la thématique du *Web mining* et du traitement de grands volumes de données. Les approches d'analyse privilégiées par *Cogniteev* sont celles basées sur l'apprentissage automatique et l'exploitation des ressources du web sémantique. L'équipe *T2I* du LIUPPA, quant à elle, travaille sur des corpus de documents textuels et s'appuie sur des systèmes d'annotation ouverts et adaptables pour la conception de systèmes

1. <https://www.nouvelle-aquitaine.fr/>

2. <http://www.cogniteev.com/>

3. <http://liuppa.univ-pau.fr/fr/index.html>

d'extraction et de recherche d'information spatiale, temporelle et thématique.

L'objectif de cette collaboration est de mettre au point un socle scientifique et technologique robuste, offrant des services dédiés au traitement de corpus textuels hétérogènes non ou peu structurés provenant du web. Il s'agit de proposer des services d'extraction et de recherche d'entités de type *entreprise* et *événement* notamment. C'est dans ce cadre que s'inscrit notre projet de thèse, qui a été baptisé « *Cognisearch* ».

1.2 Contexte de la thèse

Les récents développements des nouvelles technologies de l'information et de la communication ont fait du web une véritable mine d'information. Ces informations sont alimentées par de plus en plus de contributeurs et sont consultables par un grand nombre d'utilisateurs, grâce au développement et la facilité d'accès à Internet. Cependant, s'il peut sembler simple à un humain d'analyser et de comprendre le contenu du web, cette analyse peut s'avérer très compliquée pour une machine. Ceci s'explique sans doute par l'hétérogénéité de ce contenu. En effet, le contenu du web présente des caractéristiques particulières :

- plusieurs formats et types de données sont utilisés pour publier les informations sur le web. Les pages web sont le plus souvent écrites en HTML⁴, qui est un langage défini par le W3C⁵. D'un point de vue sémantique, les pages web sont très peu structurées dans le sens où, les informations qu'elles contiennent sont rarement mises en évidence par des balisages particuliers. De même, le contenu du web peut être de type divers et varié, nous avons notamment du texte, des images, des vidéos ;
- le contenu du web est multilingue. La démocratisation de l'accès à Internet permet à l'utilisateur de publier du contenu dans la langue de son choix. Ainsi, il est donc important de détecter la langue utilisée sur un site web, avant de commencer à l'analyser [68] ;
- le volume d'informations sur le web est gigantesque et connaît une croissance exponentielle. D'après le journal *CNRS*⁶ N°28 de janvier 2013, Twitter⁷ génère à cette époque, un flux moyen de 7 téraoctets quotidiennement et Facebook⁸ en génère 10. Ces chiffres ont connu depuis lors une forte croissance au regard de la multiplication des datacenters construits par ces entreprises.

Au regard des caractéristiques présentées ci-dessus, il est difficile de traiter automatiquement le contenu du web, en vue d'en extraire des informations pertinentes pour une tâche ciblée. C'est pourquoi les travaux s'inscrivant dans la thématique de l'extraction d'information [28] sur Internet sont en forte croissance. Une fois extraites, ces informations sont structurées et stockées dans des index. Ceux-ci sont ensuite interrogés pour répondre à des besoins d'information. La thématique associée à cette tâche est la recherche d'information (RI) [8]. Notre travail de thèse se situe à la croisée de ces deux thématiques : l'extraction d'information et la recherche d'information.

Les informations contenues sur le web renvoient le plus souvent à des entités du monde réel. Nous avons par exemple les personnes, les lieux, les entreprises ou même les événements. Ces entités sont désignées dans la littérature par l'expression : *entités nommées* (EN) [69]. Certaines EN comme les entreprises peuvent être représentées par une liste de propriétés, qui elles-mêmes peuvent correspondre à du texte (le

4. <https://www.w3schools.com/html/>

5. World Wide Web Consortium <https://www.w3.org/>

6. <http://www.cnrs.fr/fr/pdf/cim/CIM28.pdf>

7. <https://twitter.com/>

8. <https://www.facebook.com/>

nom de l'entreprise), ou à d'autres entités nommées (l'adresse du siège social). Nous appelons ce type d'entités des *EN complexes*.

Dans le cadre de notre travail, nous nous intéressons aux pages web, écrites au format HTML et plus particulièrement à leur contenu textuel. Notre objectif principal est de mettre au point des stratégies génériques permettant d'extraire des EN complexes de type *entreprise* ou *événement* à partir du texte de ces pages, de les indexer puis de les interroger pour répondre à des besoins d'informations portant sur ces entités.

1.3 Problématique

Le projet *Cognisearch* vise le développement de services d'extraction et de recherche d'information, ciblant des EN complexes. Dans le cadre de la thèse, nous ne traitons que des EN *entreprise* et *événement*. Les deux services associés ont été nommés respectivement *Cognisearch Business* et *Cognisearch Event*.

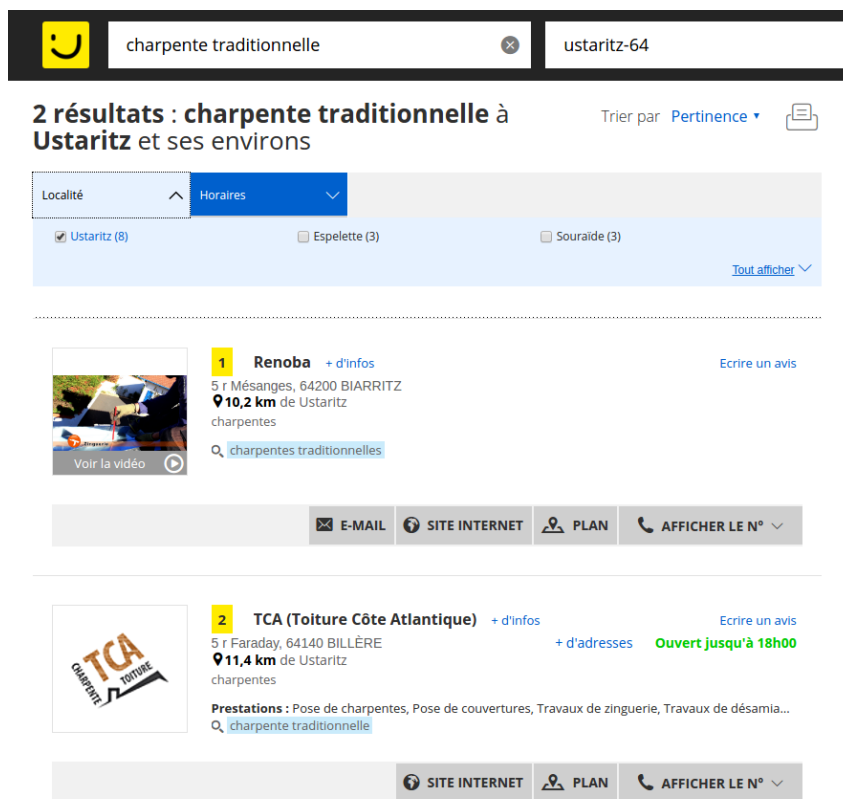


FIGURE 1.1 – Résultats de la requête « charpente traditionnelle à Ustaritz » sur les Pages Jaunes

Concernant *Cognisearch Business*, plusieurs services similaires existent sur le web. Les plus connus sont notamment Google My Business⁹, Google Maps¹⁰ ou encore les Pages Jaunes¹¹. Notons cependant que ces services sont alimentés en grande partie par des entrées manuelles des utilisateurs ou des employés. De ce fait, certaines entreprises peuvent ne pas y être répertoriées. De même celles qui y sont présentes,

9. <https://www.google.fr/business>

10. <https://www.google.fr/maps>

11. <https://www.pagesjaunes.fr/>

peuvent avoir des informations obsolètes ou manquantes. A titre d'illustration, une recherche des entreprises spécialisées dans la « charpente traditionnelle » dans la commune d'« Ustaritz » sur Google Maps ne donne aucun résultat. Avec les Pages Jaunes par contre, les deux résultats les plus pertinents obtenus sont situés à plus **10 km** de la commune ciblée (voir figure 1.1). Cependant, il existe bien une entreprise spécialisée dans la « charpente traditionnelle », localisée dans la commune d'« Ustaritz », et qui possède un site web¹² à jour avec toutes les informations la concernant. La figure 1.2 présente un extrait de la page d'accueil de ce site, elle fait bel et bien mention de « charpente traditionnelle » comme une de ses spécialités. On y voit également apparaître d'autres propriétés de l'entreprise comme, l'adresse, le numéro de téléphone ainsi que ses autres activités. Cette entreprise est bien répertoriée dans les différents services (Google Maps et Pages Jaunes), sauf qu'au moment de son enregistrement manuel, la « charpente traditionnelle » n'a pas été spécifiée comme une de ses spécialités. C'est la principale raison pour laquelle elle n'est pas retournée alors qu'elle est bien pertinente.

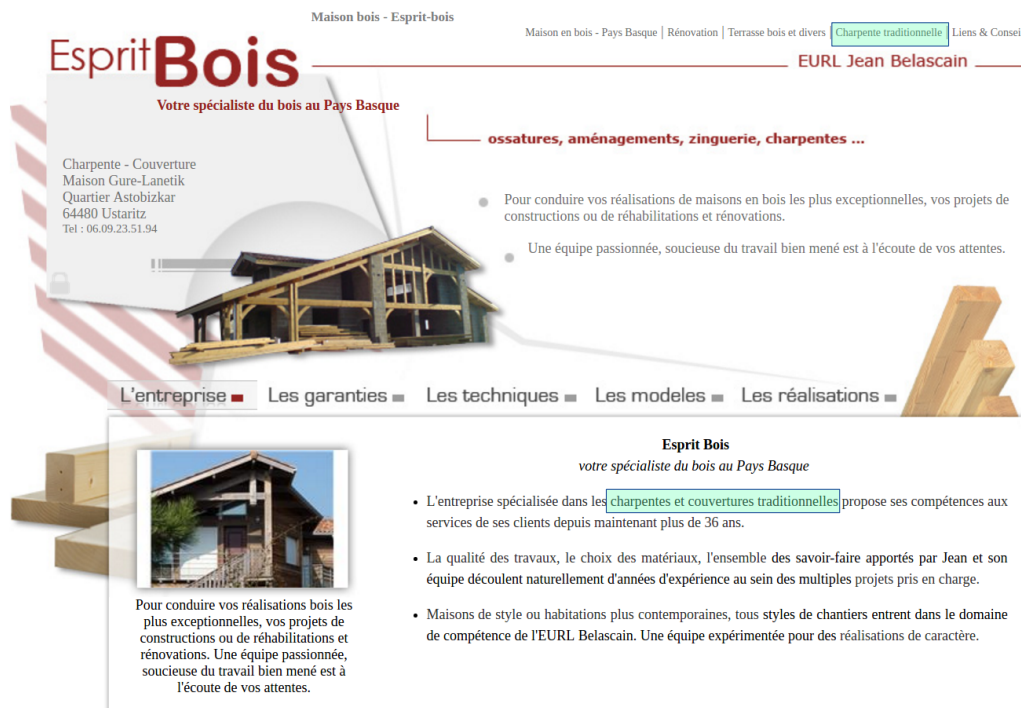


FIGURE 1.2 – Page d'accueil du site web d'une entreprise spécialisée dans la « charpente traditionnelle » à Ustaritz »

Ce sont les limites évoquées ci-dessus qui ont motivé la commande du service *Cognisearch Business*. Il est important de mentionner que le service *Cognisearch Business* n'a pas pour objectif de concurrencer les services similaires existants, mais de les compléter en exploitant les informations disponibles sur le web. En effet, nous envisageons d'extraire les données d'entreprises contenues sur le web pour alimenter notre service. Une des difficultés de cette extraction résulte du fait que les informations auxquelles nous nous

12. <http://www.esprit-bois.fr/Default.aspx>

intéressons sont noyées dans le texte des pages. Pour l'entreprise dont la figure 1.2 présente un extrait de la page d'accueil, le syntagme « charpentes et couvertures traditionnelles » qui spécifie une ses activités est complètement noyé dans le texte.

En ce qui concerne le service *Cognisearch Event*, sa demande est motivée par un constat : de plus en plus de plateformes sur le web sont mises en service pour répertorier des événements (concerts, rencontres sportives, forums économiques, meetings politiques, etc.), qui sont ensuite proposés aux utilisateurs en réponse à leurs requêtes. Ces multiples plateformes peuvent être complémentaires dans la mesure où certaines peuvent contenir des informations permettant d'enrichir celles des autres et réciproquement. La figure 1.3 illustre le cas d'un même événement répertorié sur deux plateformes distinctes du web (la première correspond à infoconcert.com et la deuxième à ticketmaster.com). L'EN complexe correspondant à cet événement est caractérisée par plusieurs propriétés : (i) le *titre* qui n'est pas tout à fait le même sur les deux plateformes ; (ii) la date qui est donnée avec précision sur la première plateforme (début et fin) ; (iii) le lieu de l'événement, qui, sur la première plateforme est renseigné par le nom de la ville, et dans la deuxième est donné avec plus de précision (l'adresse) ; (iv) la liste des intervenants (Major Laser, Radiohead, etc.) ; (v) la catégorie (Festival Musique) qui est mise en évidence sur la deuxième plateforme et pas sur la première.

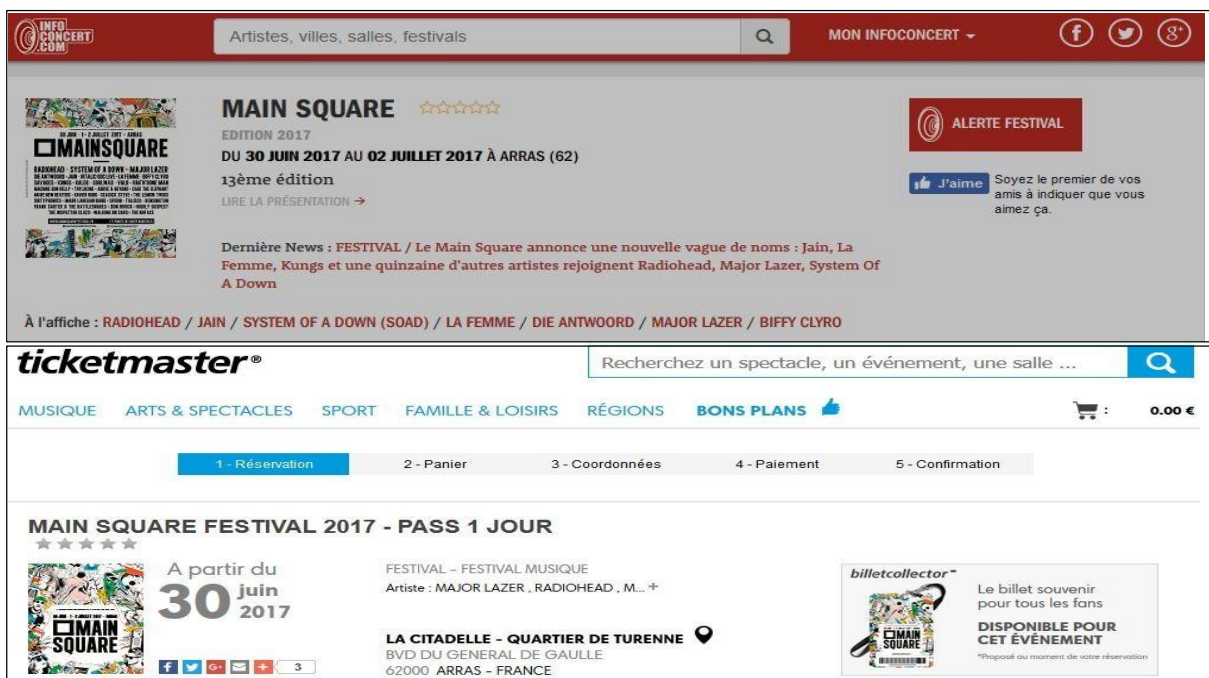


FIGURE 1.3 – Evénements similaires sur deux plateformes distinctes

L'objectif visé à travers ce deuxième service est d'analyser différentes sources du web traitant d'événements (sites de billetterie en ligne, sites de collectivités locales, sites d'offices du tourisme, annuaires, etc.) pour extraire les informations qui alimenteront la recherche d'information.

En somme, le cahier des charges des deux services demandés par *Cogniteev* soulève trois problèmes majeurs. Le premier est lié à la représentation des EN complexes. C'est cette représentation qui permet de définir explicitement de quoi est constituée une telle EN. Le deuxième est relatif à l'extraction d'EN

complexes sur le web. L'enjeu ici est de traiter le contenu du web de façon automatique pour en extraire des EN. Le troisième est lié au calcul de similarité entre EN complexes. Ce calcul est nécessaire pour garantir la cohérence de la base d'EN constituée et dans les stratégies de RI pour apparier les requêtes aux EN extraites du web.

En ce qui concerne l'extraction d'EN complexes dans les textes provenant du web, plusieurs difficultés sont à prendre en considération dans notre contexte :

- plusieurs informations présentes dans le texte pourraient correspondre à une propriété d'EN : on dit que la propriété est bruitée. C'est le cas, par exemple, du texte décrivant un événement pour lequel on peut avoir deux dates : la date de l'événement et celle de mise en vente des billets. Nous avons également le cas où la page décrivant un spectacle fait référence à l'artiste invité en même temps qu'à son manager et/ou son producteur. L'artiste, le manager et le producteur sont des EN de type *personne* et pourtant seul l'artiste ici intervient dans le spectacle. Lorsque l'extraction d'une EN complexe porte sur des textes dans lesquels certaines propriétés peuvent être bruitées, nous disons que l'extraction se fait dans une contexte bruité ;
- plusieurs propriétés d'EN complexes ne suivent pas des formes régulières en plus du fait qu'elles sont noyées dans le texte. C'est le cas par exemple du titre d'un événement qui peut être un mot (« Soprano »), un groupe nominal (« Main Square Festival 2017 ») ou même toute une phrase (« Romanès cirque tsigane : les Nomades tracent les chemins du ciel ! ») ;
- certaines propriétés d'EN complexes sont renseignés dans le texte par des syntagmes, qui eux même peuvent faire référence à d'autres EN : c'est la sur-composition. Prenons par exemple le lieu de l'événement « *Main Square Festival 2017* » sur la plateforme *ticketmaster.com* (voir figure 1.3) : *La Citadelle - Quartier de Turenne, Bvd du Général de Gaulle 62000 Arras*. Ce syntagme comprend plusieurs autres qui correspondent à des EN : (i) *La Citadelle*¹³ renvoie à une EN spatiale correspondant au nom d'un monument historique ; (ii) *Quartier de Turenne* spécifie le quartier dans lequel est localisé le monument ; (iii) *Bvd du Général de Gaulle 62000 Arras* donne l'adresse du monument et lui même contient le nom de la commune *Arras* ; La difficulté est d'extraire le syntagme correspondant au lieu dans sa totalité en tenant compte de cette sur-composition ;

Pour évaluer la similarité entre EN complexes, l'approche « classique » consiste à comparer les propriétés entre elles, puis les différents scores obtenus sont combinés pour en déduire la similarité globale. Dans notre contexte, ce calcul pose des problèmes :

- au niveau du calcul de la similarité entre propriétés, la première difficulté découle du fait que les propriétés ne sont pas toujours exprimées de la même façon. C'est le cas des lieux des événements de la figure 1.3, ils n'ont pas le même niveau de détail : l'un est donné par une adresse et l'autre par un nom de ville. De même, certaines propriétés peuvent être multivaluées : c'est le cas par exemple, des intervenants dans les événements de la figure 1.3 qui sont des ensembles de noms d'artistes et de groupes ;
- au niveau de l'EN globale, nous devons tenir compte d'une difficulté majeure : certaines propriétés d'EN complexes peuvent être non renseignées. En reprenant les deux événements de la figure 1.3, on se rend compte que la catégorie de l'événement est mise en évidence sur la deuxième plateforme et pas sur la première. De ce fait, il n'est pas possible d'évaluer la similarité entre les catégories si celle-ci n'est pas détecté pour une des EN *événement*. L'enjeu pour nous est de pouvoir évaluer la similarité globale en prenant en considération ces cas de figure.

13. <https://www.lillelanuit.com/annuaires/lieux/la-citadelle-darras/>

Par ailleurs, la réalisation de ces services devra se faire sous certaines contraintes :

- les approches d'extraction d'EN complexes à mettre en œuvre doivent être indépendantes du type de celles-ci (généricité). Il en est de même pour les approches de calcul de similarité ;
- les ressources utilisées pour l'extraction d'EN doivent être libres d'accès (non propriétaires) et sans quotas.

1.4 Contributions

Nos travaux visent deux contributions relatives à l'extraction d'EN complexes sur le web et au calcul de similarité. En ce qui concerne l'extraction d'EN complexes sur le web, nous envisageons la conception d'une chaîne générique permettant de traiter de la même façon l'extraction d'EN complexes en général, et celle de type *entreprise* et *événement* en particulier. Cette chaîne devra prendre en compte des différents verrous propres à notre contexte. Il s'agit notamment d'extraction d'information dans un contexte bruité, de formes d'expressions non régulières, ou encore de sur-composition.

La seconde contribution, quant à elle, porte sur le calcul de similarité entre EN complexes. Nous envisageons des approches génériques, valides pour tout type d'EN et qui tiennent compte des différentes formes de représentation de propriétés. Le principal verrou consiste à évaluer la similarité entre EN complexes dont certaines propriétés peuvent ne pas être renseignées.

Nos travaux ciblent également la définition de modèles de représentation d'EN complexes. Nous reprenons des modèles existants que nous complétons de façon originale pour représenter les informations extraites du web.

1.5 Organisation du mémoire

La suite du mémoire s'organise en trois parties. La première est dédiée à l'état de l'art dans lequel nous faisons un tour d'horizon des principaux axes de recherche explorés dans le cadre de la thèse. Cette partie s'organise en six chapitres. Le premier rappelle notre problème et les axes de recherche abordés par celui-ci. Le deuxième définit la notion d'« entité nommée » qui est au cœur de notre travail. Les trois chapitres suivants dressent un état de l'art pour chacun des axes traités dans la thèse : modélisation des EN, extraction d'EN sur le web et calcul de similarité entre EN. Dans le dernier chapitre, nous discutons de la pertinence des travaux de l'état de l'art pour notre problématique.

La deuxième partie du mémoire présente nos contributions et elle est constituée de cinq chapitres. Le premier présente l'architecture du socle que nous avons conçu pour répondre à notre problématique. Le deuxième, présente les modèles que nous avons définis ou ré-utilisés pour représenter certaines EN notamment les adresses, les entreprises et les événements. Le troisième chapitre présente notre proposition pour l'extraction d'EN complexes dans un texte. Le quatrième chapitre détaille nos propositions pour le calcul de similarité entre EN complexes. Le dernier chapitre présente l'implémentation de ces propositions dans les prototypes des services *Cognisearch Business* et de *Cognisearch Event*.

La dernière partie conclut le mémoire en rappelant les problèmes et nos propositions. Nous y présentons aussi les perspectives ouvertes par nos travaux.

Deuxième partie

État de l'art : de l'extraction à la recherche d'entités nommées

Chapitre 2

Contexte et problèmes de recherche

L'objectif du projet *Cognisearch* (figure 2.1) est d'analyser le contenu du web pour extraire des EN complexes notamment des entreprises et des événements. Ces EN complexes seront ensuite stockées dans des index pour alimenter des services de recherche d'information. Deux questions majeurs émergent : (i) comment traiter le contenu du web pour extraire ces EN ? (ii) comment trouver les EN pertinentes pour une requête donnée ? Ces deux questions soulèvent plusieurs problèmes :

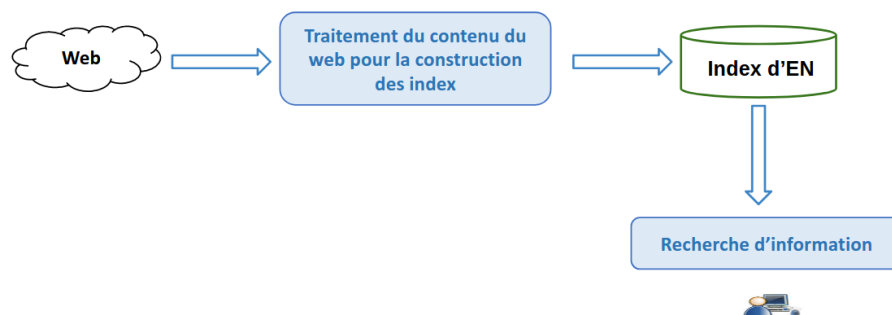


FIGURE 2.1 – Vue simplifiée du projet *Cognisearch*

- le premier est relatif à l'identification du contenu du web pertinent pour une tâche précise d'extraction. En effet, la masse de données disponible sur le web est gigantesque¹⁴ et s'évalue en Zettaoctets¹⁵. De ce fait, lancer un processus d'analyse à l'échelle du web tout entier serait une opération très gourmande en temps et en ressources. D'où l'enjeu de prétraiter le web pour cibler l'extraction sur un sous-ensemble précis de son contenu ;
- le deuxième est lié à l'extraction d'information dans les pages web. Pour notre problème, nous ciblons le contenu textuel de ces pages. Notre processus d'extraction doit donc prendre en compte l'hétérogénéité des informations contenues dans les textes analysés. Il peut s'agir notamment : (i) d'informations spatiales (noms de lieux, adresses, villes, ...); (ii) d'informations temporelles (dates, horaires, durées, ...); (iii) ou même d'informations thématiques (activités d'entreprises, noms de

14. <https://www.waterfordtechnologies.com/big-data-interesting-facts/>

15. 1 Zo = 10²¹ octets

- personnes ou d’organisations, catégories d’événements, ...);
- le troisième est relatif à la structuration et l’organisation des informations. Les EN complexes ont la particularité d’être composées de plusieurs propriétés, dont certaines peuvent être d’autres EN. L’enjeu pour nous est de les représenter convenablement à l’aide de modèles adéquats. L’extraction d’information s’appuie également sur ces modèles pour définir les informations à cibler dans le texte;
- le quatrième est lié au calcul de similarité entre EN complexes. Ce calcul est nécessaire lors de la construction des index, pour garantir l’intégration sans doublon des EN extraites. De même, la phase de recherche d’information exploite également des fonctions de calcul de similarité pour appairer les besoins d’information aux EN indexées.

La figure 2.2 décrit l’enchaînement des traitements que nous envisageons pour adresser notre problématique. Les différents problèmes soulevés s’articulent autour de trois principaux axes de recherche : (i) les modèles de représentation d’EN. Ce sont ces modèles qui définissent les descripteurs d’EN ; (ii) l’extraction d’EN à partir du contenu du web. Cet axe englobe les prétraitements nécessaires pour cibler le contenu du web à traiter, ainsi que le processus d’extraction à mettre en œuvre pour identifier les descripteurs d’EN dans les textes correspondants ; (iii) le calcul de similarité entre EN. Ce calcul sert à garantir la cohérence de l’index et à appairer les besoins d’information aux EN de l’index. Dans cette partie, nous dressons un état de l’art de nos connaissances pour chacun de ces axes. Avant cela, un premier chapitre est consacré à la définition de la notion *d’entité nommée*. Enfin, dans le bilan général, nous discutons de l’intérêt des travaux présentés et de leur potentielle intégration dans nos contributions envisagées.

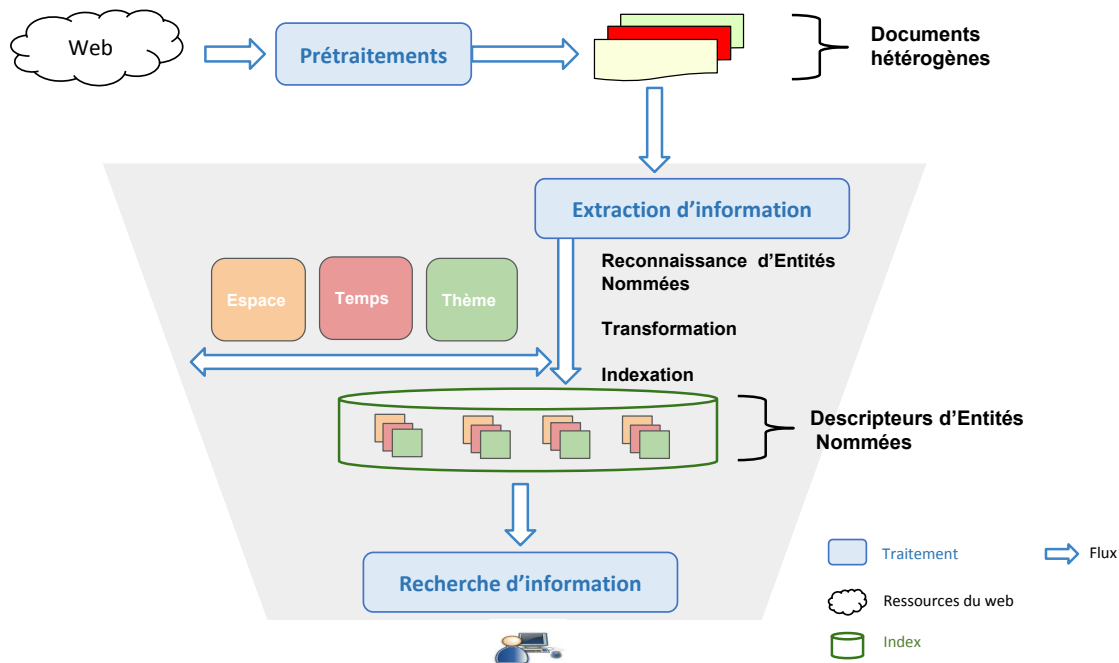


FIGURE 2.2 – La solution envisagée pour le projet *Cognisearch*

Chapitre 3

Entité Nommée : définition et catégories

La notion d'entité nommée (EN) a fait son apparition dans le cadre des conférences MUC¹⁶ (*Message Understanding Conference*) [69], organisées conjointement par plusieurs institutions américaines et financées par la DARPA¹⁷ (*Defense Advanced Research Projects Agency*). L'objectif de ces conférences était de promouvoir les travaux de recherche autour de l'extraction d'information en général et dans le domaine militaire en particulier.

La notion d'EN a beaucoup évolué au fil des années, même si une définition standard ne s'est pas encore imposée [97]. La plupart des définitions proposées se contentent d'énumérer les types d'EN d'un point de vue « informationnel ». Le NIST¹⁸ (*National Institute of Standards and Technology*) propose, par exemple, la définition suivante :

« *Named Entity : a named object of interest such as a person, organization, or location.* »

D'autres définitions, spécifiques au domaine du Traitement Automatique du Langage (TAL), sont beaucoup plus orientées linguistique. Par exemple, Vicente [145] propose la définition suivante :

« *Entité Nommée est la notion utilisée en TAL pour désigner les éléments discursifs mono-référentiels qui coïncident en partie avec les noms propres (note : la notion d'entité nommée concerne les noms propres mais aussi les dates et les mesures) et qui suivent des patrons syntaxiques déterminés.* »

Une définition qui nous semble plus « complète » est celle proposée dans l'encyclopédie Atalapédia de l'ATALA¹⁹ (Association pour le Traitement Automatique des Langues) et reprise dans la thèse de Maud Ehrmann [47] :

« *Les entités nommées désignent l'ensemble des noms de personnes, de lieux, d'entreprises, etc. contenues dans un texte. On ajoute souvent à ces éléments les dates et d'autres données chiffrées. Par extension, les entités désignent parfois les éléments de base pour une tâche donnée (par exemple, les noms de gènes dans le cadre de l'étude des textes de biologie). [...] Ces séquences*

16. http://www.itl.nist.gov/iaui/894.02/related_projects/tipster/muc.htm

17. <http://www.darpa.mil/>

18. http://www.itl.nist.gov/iad/894.02/related_projects/muc/info/definitions.html

19. <http://www.atala.org/>

référentielles sont primordiales pour beaucoup d'applications linguistiques, que ce soit la recherche ou l'extraction d'information, la traduction automatique ou la compréhension de textes. »

En somme, une EN peut donc être définie comme une unité linguistique (syntagme), identifiable de façon unique dans un contexte précis et qui renvoie à un objet du monde réel [45]. Dans ce travail, nous traiterons les EN en les représentant en deux principales catégories : les EN « élémentaires » et les EN « complexes »

Les EN élémentaires

Une EN élémentaire est représentée par un syntagme qui permet d'identifier de façon unique l'objet auquel elle fait référence. Les EN énumérées dans les diverses définitions proposées dans l'état de l'art font partie de cette catégorie. Une EN élémentaire peut donc être :

- une personne (ex : « Isaac Newton », « Victor Hugo »);
- une organisation (ex : « UNICEF », « Google »);
- un lieu (ex : « Le Bataclan », « Pau »);
- une date (ex : « 13 Janvier 2015 », « printemps 2017 »);
- une information de contact (ex : un e-mail « jean.dujardin@univ-pau.fr »).

Les EN complexes

Une EN complexe est une EN représentée par une liste de *propriétés*. Par exemple, une EN correspondant à un film peut être décrite par 4 propriétés :

- **titre** : « Star Wars »;
- **date de sortie** : « 25 Mai 1977 »;
- **réalisateur** : « Georges Lucas »;
- **catégories** : « Action », « Aventure », « Science-fiction ».

Dans cet exemple, le film est traité comme une EN complexe. Le *titre* est un texte, le *réalisateur* est une EN élémentaire de type *personne*, et les *catégories* sont représentées par un ensemble de textes.

Il est important de noter que la catégorie d'une EN dépend fortement du contexte. En effet, une même EN peut être vue comme une EN élémentaire ou une EN complexe. Par exemple, une EN *personne* peut être élémentaire si on la représente uniquement par son nom, ou complexe si en plus de son nom, on a une date et un lieu de naissance. Ainsi, dans l'exemple précédent, le réalisateur pourrait être une EN complexe décrite par les propriétés suivantes :

- **nom** : « Georges Lucas »;
- **date de naissance** : « 14 Mai 1944 »;
- **lieu de naissance** : « Californie - USA »;

De façon plus formelle, une EN complexe e est une EN composée de n ($n > 1$) propriétés p_i ($i \in [1, n]$). En utilisant les mêmes notations que Gupta et al. [73], nous définissons e par :

$$e = \langle p_1, p_2, \dots, p_n \rangle$$

La propriété p_i peut être :

- monovaluée ;
- multivaluée.

La valeur prise par la propriété p_i peut être :

- un type simple (texte, nombre, etc.) ;
- une EN élémentaire ;
- une EN complexe.

Différents objets informationnels sont généralement traités comme des EN complexes. Il s'agit notamment :

- des points d'intérêts : un POI est un « point précis de la terre » qui peut s'avérer utile ou intéressant pour un individu [141]. Il s'agit par exemple d'un site touristique, d'une montagne, d'une école, ou même d'un commerce. Un POI est généralement décrit par, au moins, un nom, une catégorie et une localisation géographique ;
- des entreprises : une entreprise est une « unité institutionnelle »²⁰ avec pour but de produire des biens ou de fournir des services. C'est un cas particulier de POI avec des caractéristiques supplémentaires comme le(s) nom(s) de gérant(s), l'ensemble d'activités, ou même des informations administratives ;
- des événements : un événement est défini comme « quelque chose qui se passe quelque part »²¹. Un lieu précis, une période de temps donnée et un sujet sont nécessaires pour décrire un événement. Dans la littérature deux types d'événements sont distingués :
 - les faits qui correspondent à des informations d'actualité, à des événements historiques, ou même à des péripéties d'une narration ;
 - les événements socio-culturels qui sont des événements planifiés et auxquels un public est associé. Nous pouvons citer les concerts, les festivals, les colloques, les rencontres sportives, les meetings politiques, etc.

Nous allons nous intéresser dans le chapitre suivant aux modèles proposés dans la littérature pour la représentation d'EN.

20. <https://fr.wikipedia.org/wiki/Entreprise>

21. <http://www.larousse.fr/dictionnaires/francais/événement/31839>

Chapitre 4

Modèles de représentation d'entités nommées

La définition des modèles pour représenter les EN s'inscrit dans la thématique de la représentation de la connaissance. La plupart des modèles proposés pour représenter les EN élémentaires s'appuient sur le langage de balisage XML (eXtension Markup Language). Nous avons par exemple le standard ENAMEX²², proposé dans le cadre des campagnes MUC, qui permet de baliser des EN élémentaires dans le texte. En ce qui concerne les EN complexes, la définition de modèles pour les représenter doit prendre en compte le fait qu'elles sont constituées de plusieurs propriétés de type varié, avec d'éventuelles relations entre elles. Dans la pratique, plusieurs modèles sont parfois combinés pour représenter les EN complexes.

Ce chapitre explore les principaux modèles existants pour la représentation des EN (élémentaires et complexes). Nous les avons organisés en catégories que nous présentons dans une première section. Une deuxième section, quant à elle, énumère les modèles utilisés pour représenter différents types d'EN.

4.1 Catégories de modèles de représentation d'EN

Dans cette section, nous organisons les modèles proposés pour la représentation des EN en trois principales catégories : les modèles issus des standards du web, les modèles de type ontologique et les modèles « ad-hoc ».

4.1.1 Les modèles issus des standards du web

Dans l'optique d'introduire de la sémantique dans les pages du web, certains formalismes ont été proposés pour structurer les informations. L'idée est de mettre en évidence une information particulière en utilisant des balises et attributs qui ne sont pas interprétées par le navigateur. C'est ainsi que des formalismes comme Google Base²³, Google Data²⁴, Yahoo Common Tag²⁵ ou encore microformats.org²⁶ ont vu

22. http://cs.nyu.edu/faculty/grishman/NEtask20.book_6.html

23. http://microformats.org/wiki/Google_Base

24. <https://developers.google.com/gdata/>

25. <http://commontag.org/>

26. <http://microformats.org>

le jour au début des années 2000. En 2011, Bing, Google et Yahoo ont lancé conjointement `schema.org`²⁷ pour faciliter la compréhension de l'information par leurs robots d'indexation. Il s'agit d'un formalisme inspiré de `microformats.org` et d'autres formalismes existants, pour lequel de nombreuses extensions ont été définies. Ces formalismes sont intégrés dans des pages web en utilisant le langage XML. Le listing 4.1 ci-dessous, illustre un extrait de page qui utilise le formalisme `schema.org` pour baliser une EN correspondant à un film. L'attribut `itemprop` permet de spécifier les propriétés du film et l'attribut `itemtype` permet de spécifier le type d'une EN ou de ses propriétés. Un film, selon ce modèle, possède une propriété (*director*) qui permet de spécifier le réalisateur. Pour notre exemple, le réalisateur est une EN de type *personne* avec un nom et une date de naissance (voir lignes 3 à 6).

```

1 <div itemscope itemprop="http://schema.org/Movie">
2   <h1 itemprop="name">Avatar</h1>
3   <div itemprop="director" itemscope itemtype="http://schema.org/Person">
4 Réalisateur : <span itemprop="name">James Cameron</span>
5     (né le <time itemprop="birthDate" datetime="1954-08-16">16 aout 1954</time>)
6   </div>
7   <span itemprop="genre">Science-fiction</span>
8   <a href=" ../movies/avatar-theatrical-trailer.html" itemprop="trailer">Bande-annonce
9     </a>
10 </div>

```

Listing 4.1 – Représentation d'un film suivant le formalisme `schema.org`

4.1.2 Les modèles de type ontologique

Plusieurs définitions sont proposées dans la littérature pour la notion d'ontologie. Une des plus utilisées est celle proposée par Gruber [70] :

« *An ontology is an explicit specification of a conceptualization.* »

Hammache et al. [76] identifient deux principales notions dans cette définition de l'ontologie : (i) la notion de conceptualisation qui renvoie à une modélisation abstraite d'un phénomène réel, ceci en se basant sur l'identification de concepts pertinents pour décrire ledit phénomène ; (ii) la notion de formalisation pour faciliter le partage de la connaissance ou pour permettre son exploitation automatique par une machine.

De même, d'autres définitions mettent un accent particulier sur la notion de relations entre termes et concepts dans la représentation de la connaissance. Neches et al. [127] proposent la définition suivante :

« *An ontology defines the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary* »

En somme, une ontologie permet de conceptualiser de façon formelle la connaissance d'un domaine, en décrivant explicitement les concepts et les relations entre eux. Par conséquent, cette notion est très utilisée pour la représentation de différents types d'EN propres à des domaines précis. Cette tendance s'est généralisée avec l'apparition du web sémantique et le besoin de partager la connaissance sur internet. Nous avons, par exemple, l'ontologie MESH [18] pour la modélisation des EN du domaine biomédical, ou encore l'ontologie PLANTS [67] pour la représentation des EN du domaine botanique.

27. <http://schema.org>

4.1.3 Les modèles « ad-hoc »

Certains travaux de recherche proposent des modèles dédiées à la représentation des EN. Ces modèles sont très souvent proposés à des fins d’extraction d’information et permettent de représenter les EN sous une forme qui facilite leurs traitements. Dans cette catégorie nous avons également les modèles proposés par des organismes publics, pour uniformiser la représentation et le partage de l’information. Parmi ces organismes, nous avons, par exemple, l’Etalab²⁸, l’INSEE²⁹ ou encore Pôle Emploi³⁰.

4.2 Exemples de modèles de représentation d’EN

Nous explorons quelques modèles utilisés pour la représentation de différents types d’EN : les EN *temporelles*, les EN de type *lieu*, les EN *sociales* (personne et organisation), les EN *entreprise* et les EN *événement*. Nous considérerons deux EN complexes qui nous serviront d’exemples tout au long de la présentation des modèles :

- l’entreprise Cogniteev ;
- l’événement « *Tournée d’Eric Clapton en France avec deux prestations : la première le 24 Novembre 2017 à 21h00 au Bataclan avec Peter Green ; et la deuxième au Zénith de Lille le 28 Novembre 2017 à 21h30 en compagnie de Scott Henderson* ».

4.2.1 Modèles de représentation des EN temporelles

TIMEX3 est le modèle de représentation des EN temporelles le plus répandu. Il permet d’associer des représentations aux informations temporelles dans le format standard ISO-8601. Le langage TimeML [138], par exemple, utilise le modèle TIMEX3 pour le marquage de l’information temporelle dans le texte. Le listing 4.2 illustre un fragment de texte dont les EN temporelles sont balisées en TimeML.

```

1 la première le <TIMEX3 tid="t1" type="DATE" value="2017-11-24"> 24 Novembre 2017 </
  TIMEX3> à <TIMEX3 tid="h1" type="TIME" value="T21:00"> 21h00 </TIMEX3> au
2 Bataclan avec Peter Green.
```

Listing 4.2 – Représentation d’EN temporelles en TimeML

4.2.2 Modèles de représentation des EN spatiales

En ce qui concerne la représentation des lieux, plusieurs modèles existent également. Certains sont des standards du web, d’autres des ontologies et d’autres encore des modèles ad-hoc.

- La classe **Place**³¹ de schema.org : un lieu selon ce modèle contient plusieurs propriétés, notamment son nom et sa localisation. Dans ce modèle, il est possible de spécifier la représentation des lieux d’un point de vue géométrique. Ainsi, lorsque le lieu est un point, on lui associe un objet

28. Laboratoire gouvernemental français : <https://www.etalab.gouv.fr>

29. Institut National de la Statistique et des Études Économiques : <https://www.insee.fr/fr/accueil>

30. <http://www.pole-emploi.fr/accueil/>

31. <http://schema.org/Place>

GeoCoordinates³² pour spécifier ses coordonnées, dans le cas où il s'agit d'une géométrie particulière, c'est un objet GeoShape³³ qui permet de définir la forme correspondante. Un exemple de représentation de la salle de spectacle « *Bataclan* » avec ce modèle dans une page web est donnée par le listing 4.3 :

```

1   <div itemscope itemtype="http://schema.org/Place">
2     <h1 itemprop="name">Bataclan</h1>
3     <div class="address" itemprop="address" itemscope
4       itemtype="http://schema.org/PostalAddress">
5       <h3 itemprop="streetAddress"> 50 Boulevard Voltaire</h3><br>
6       <h3 itemprop="addressLocality">Paris</h3>,
7       <h3 itemprop="postalCode">75011</h3>
8       <div itemprop="geo" itemscope
9         itemtype="http://schema.org/GeoCoordinates">
10        <span itemprop="latitude" content="48.86"></span>
11        <span itemprop="longitude" content="2.37"></span>
12      </div>
13    </div>
14  </div>
15

```

Listing 4.3 – Représentation du « Bataclan » suivant schema.org/Place

- Les classes **h-geo**³⁴ et **h-adr**³⁵ de microformats.org : la classe *h-geo* permet de représenter simplement un point de l'espace dans le système WGS 84³⁶ avec sa latitude, sa longitude et son altitude. De même, la classe *h-adr* est dédié à la représentation des lieux de type adresse. Le listing 4.4 illustre la représentation du « Bataclan » selon *h-adr* dans un extrait de page web.

```

1   <div class="h-adr">
2     <h1 class="p-extended-address">Bataclan</h1>
3     <h3 class="p-street-address">50 Boulevard Voltaire</h3>
4     <h3 class="p-locality">Paris</h3>
5     <h3 class="p-country-name">France</h3>
6     <span h3="p-postal-code">75011</h3>
7     <p class="h-geo">
8       <span class="p-latitude">48.86</span>,
9       <span class="p-longitude">2.37</span>
10      <span class="p-altitude">30</span>
11    </p>
12  </div>
13

```

32. <http://schema.org/GeoCoordinates>

33. <http://schema.org/GeoShape>

34. <http://microformats.org/wiki/h-geo>

35. <http://microformats.org/wiki/h-adr>

36. https://fr.wikipedia.org/wiki/WGS_84

Listing 4.4 – Représentation du « Bataclan » suivant h-adr

- L'ontologie GeorSS³⁷ : elle permet d'associer un lieu à l'information contenue dans un flux RSS. Elle permet entre autres de représenter plusieurs géométries correspondant à une EN spatiale, ainsi que leurs relations avec d'autres EN spatiales. Cette ontologie s'appuie sur le langage de modélisation *Geography Markup Language* (GML) défini par l'Open Geospatial Consortium³⁸ (OCG). Considérons que le texte correspondant à la prestation d'Eric Clapton au Bataclan soit contenu dans un flux RSS, dont le listing 4.5 présente un extrait. Le lieu associé à cette prestation est défini par ses coordonnées géographiques et celles-ci sont contenues dans les balises **geo:lat** (ligne 8) et **geo:long** (ligne 9) de l'extrait de flux RSS :

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <rss version="2.0" xmlns:georss="http://www.georss.org/georss"
3  xmlns:gml="http://www.opengis.net/gml">
4  ...
5  <item>
6  <title> Eric Clapton au Bataclan à Paris </title>
7  ...
8  <geo:lat>4.86</geo:lat>
9  <geo:long>2.37</geo:long>
10 ...
11 </item>
12 </rss>
13
```

Listing 4.5 – Représentation du « Bataclan » suivant GeorSS

- Le modèle Adresse³⁹ de l'Etalab : il s'agit d'un modèle consensuel défini pour la représentation uniforme des adresses de la BAN⁴⁰. Ce modèle permet de définir une adresse au niveau de granularité de la voie. Les propriétés principales d'une adresse selon ce modèle sont décrites dans le tableau 4.1. Les exemples illustrant chaque propriété correspondent à la représentation de l'adresse de l'entreprise *Cogniteev* selon ce modèle.

37. <http://www.georss.org>

38. <http://www.opengeospatial.org/>

39. <https://adresse.data.gouv.fr/pdf/description-contenu.pdf>

40. Base Adresse Nationale : répertoire national des adresses françaises construit à partir des données de l'IGN et de La Poste

Propriétés	Définitions	Exemples
id	Identifiant unique de l'adresse dans la base nationale d'adresses (BAN)	ADRNIVX_0000000276372247
numero_voie	Numéro de la voie de circulation	2
nom_voie	Nom de la voie de circulation	Allée du Doyen Georges Brus
rep	Indice de répétition associé au numéro de voie (Bis, Ter, ...)	-
nom_commune	Nom officiel de la commune	Pessac
code_insee	Identifiant INSEE de la commune suivant le Code Officiel Géographique (COG)	33600
code_postal	Code postal associé à la commune	33600
nom_ld	Nom du lieu-dit associé à l'adresse	-
lon	Longitude du point géographique associé à l'adresse dans le système WGS 84 ⁴¹	-0,62
lat	Latitude du point géographique associé à l'adresse dans le système WGS 84	44,79

TABLE 4.1 – Propriétés de l'adresse de Cogniteev avec le modèle Adresse de l'Etalab

4.2.3 Modèles de représentation des EN sociales

On entend par « entité sociale » le terme renvoyant à une personne au sens juridique. Il peut s'agir d'un individu ou même d'une organisation. Il existe de nombreux modèles dédiés à la représentation de ce type d'entités.

- La classe **h-card** ⁴² de microformats.org : cette classe permet de représenter les deux types d'entités sociales (personne et organisation). Une entité sociale, selon ce modèle, est définie par plusieurs propriétés parmi lesquelles son nom et ses coordonnées (contacts et/ou de localisation). Pour le cas des individus, il est possible de spécifier les organisations auxquelles ils sont rattachés (employeur, banque, etc.). La représentation de l'artiste *Eric Clapton* avec ce modèle dans un extrait de page web est illustré par le listing 4.6.

```

1 <div class="h-card">
2   <h3 class="p-name">Eric Clapton</h3>
3   <span class="u-email">eric.clapton@hotmail.com</span>
4   <span class="dt-bday">30 Mars 1945</span>
5   <span class="p-job-title">Guitariste</span>
6 </div>
7
```

Listing 4.6 – Représentation d'Eric Clapton selon le modèle h-card

- Les classes **Person** ⁴³ et **Organization** ⁴⁴ de schema.org : elles sont définies à partir de la classe **h-card** de microformats.org. La classe **Person** est dédiée à la représentation des individus et permet

42. <http://microformats.org/wiki/hcard-fr>

43. <http://schema.org/Person>

44. <http://schema.org/Organization>

également de représenter des personnages de fiction. La classe **Organization**, quant à elle, permet de représenter les organisations comme les écoles, les équipes sportives, les institutions publiques, etc. Chaque instance de ces classes est définie par une liste de propriétés parmi lesquelles le nom, la localisation, les alias éventuels, les informations de contact, ainsi que les relations entre instances (appartenance, inclusion, etc.). Le listing 4.7 illustre un extrait de page web dans laquelle l'entreprise *Cogniteev* est représentée avec la classe **Organization**. Cette représentation tient compte du fait qu'un individu (Armel Fotsoh) en soit membre.

```

1  <div itemscope itemtype="http://schema.org/Organization">
2    <span itemprop="name">Cogniteev SAS</span>
3    <div itemprop="address" itemscope
4      itemtype="http://schema.org/PostalAddress">
5      <span itemprop="streetAddress">2 Allée du Doyen Georges Brus</span>
6      <span itemprop="postalCode">33600</span>
7      <span itemprop="addressLocality">Pessac, France</span>
8    </div>
9
10   Tel : <span itemprop="telephone">09 72 53 75 22 </span>,
11   E-mail: <span itemprop="email">hello(at)cogniteev.com</span>
12
13   <div itemprop="member" itemscope itemtype="http://schema.org/Person">
14     <span itemprop="name">Armel Fotsoh</span>
15     <span itemprop="jobTitle">Doctorant</span>
16   </div>
17 </div>
18

```

Listing 4.7 – Représentation l'entreprise Cogniteev avec la classe schema.org/Organization

- Les classes **foaf:Person** et **foaf:Organization** de l'ontologie FOAF⁴⁵ : FOAF est une ontologie définie à la base pour décrire les personnes ainsi que les relations qu'elles entretiennent. Elle est une référence dans le domaine du web sémantique et s'appuie sur la syntaxe RDF⁴⁶. La classe **foaf:Person** est dédiée à la représentation des individus. Elle permet de spécifier plusieurs propriétés d'un individu, dont son identité ainsi que ses relations avec d'autres. La classe **foaf:Organization**, quant à elle, permet de représenter des groupes d'individus. L'entreprise *Cogniteev* représentée suivant ce modèle est illustrée par le listing 4.8 :

```

1  <foaf:Organization>
2    <foaf:name>Cogniteev SAS</foaf:name>
3    <foaf:member>
4      <foaf:Person>
5        <foaf:name> Armel Fotsoh </foaf:name>
6      </foaf:Person>
7    </foaf:member>

```

45. Friend of a Friend <http://xmlns.com/foaf/spec/>

46. <https://www.w3.org/RDF/>

```

8   </foaf:Organization>
9

```

Listing 4.8 – Représentation l'entreprise Cogniteev avec la classe foaf:Organization

On constate que, contrairement à la classe **Organization** de schema.org, la classe **foaf:Organization** ne permet pas de représenter les informations spatiales d'une organisation.

4.2.4 Modèles de représentation d'EN *entreprise*

En ce qui concerne la représentation de EN *entreprise*, nous nous intéressons à deux modèles en particulier : la classe **LocalBusiness** de schema.org et le modèle **SIRENE** de l'INSEE.

- La classe **LocalBusiness**⁴⁷ : c'est une spécialisation des classes **Organization**⁴⁸ et **Place**⁴⁹, pour spécifier les informations d'une entreprise ou celles d'une de ses succursales. Ainsi, en plus des propriétés relatives à l'identité de l'entreprise (nom, raison sociale, etc.), nous avons également les informations de localisation de celle-ci ainsi que les horaires et jours d'ouverture. Des classes spécialisées permettent de représenter certaines entreprises, en fonction de leur activité (*Dentist*, *Store*, *TravelAgency*, etc.). Pour le cas de l'entreprise *Cogniteev*, sa représentation avec ce modèle est illustrée par listing 4.9.

```

1   <div itemscope itemtype="http://schema.org/LocalBusiness">
2     <h3 itemprop="name">Cogniteev SAS</h3>
3     <div itemprop="address" itemscope
4       itemtype="http://schema.org/PostalAddress">
5       <span itemprop="streetAddress">2 Allée du Doyen Georges Brus</span>
6       <span itemprop="postalCode">33600</span>
7       <span itemprop="addressLocality">Pessac, France</span>
8
9       <div itemprop="geo" itemscope
10        itemtype="http://schema.org/GeoCoordinates">
11        <span itemprop="latitude" content="44.79"></span>
12        <span itemprop="longitude" content="-0.62"></span>
13      </div>
14    </div>
15
16    Tel : <span itemprop="telephone">09 72 53 75 22 </span>,
17    E-mail: <span itemprop="email">hello(at)cogniteev.com</span>
18
19    <div itemprop="member" itemscope
20      itemtype="http://schema.org/Person">
21      <span itemprop="name">Armel Fotsch</span>
22      <span itemprop="jobTitle">Doctorant</span>
23    </div>

```

47. <http://schema.org/Place>

48. <http://schema.org/Organization>

49. <http://schema.org/Place>

```

24
25   Horaires d'ouverture :
26   <h3 itemprop="openingHours" content="Mo-Fri 08:00-18:30"> De lundi
27   à vendredi, entre 8h00 et 18h30</h3>
28 </div>
29

```

Listing 4.9 – Représentation l'entreprise Cogniteev avec la classe schema.org/LocalBusiness

- Le modèle **SIRENE** : c'est un modèle ad-hoc, qui est utilisé pour représenter les entreprises et les établissements en France dans la base de données du système national d'identification des entreprises et établissements⁵⁰ (SIRENE). Le tableau 4.2 dresse la liste des propriétés de base d'une entreprise selon ce modèle. Notons que ce modèle donne également la possibilité de représenter des informations plus spécifiques comme : le nombre de salariés, le chiffre d'affaires, les statistiques déclarées de l'entreprise à des dates données, etc.

Propriétés	Types	Exemples
Nom officiel	Chaîne de caractères	Cogniteev SAS
Nom commercial	Chaîne de caractères	Cogniteev
Identifiant (Numéro SIRENE)	Entier	792 261 794
Statut juridique	Chaîne de caractères	Société par actions simplifiée
Siège social de l'entreprise	Chaîne de caractères	2 Allée du Doyen Georges Brus 33600 Pessac
Capital social	Réel	113 235,00
Liste des dirigeants	Liste de chaîne de caractères	[François Goube]
Code APE (activité principale)	Chaîne de caractères	6201Z

TABLE 4.2 – Propriétés principales d'une entreprise selon le modèle SIRENE

4.2.5 Modèles de représentation d'EN *événement*

De nombreux modèles existent pour représenter les événements. Nous avons des ontologies comme **EVENT**, **LODE** ou **SEM**, des modèles de type standard du web (les classes **h-event** et **Event** de microformats.org et de schema.org respectivement), et des modèles ad-hoc (ACE).

- **EVENT** [143] : il s'agit d'une ontologie développée à l'université de Londres. Elle suit les standards du W3C et est réputée simple. La figure 4.1 présente un extrait de la représentation de l'événement relatif à la tournée d'Eric Clapton, selon le modèle EVENT. La tournée est décrite comme un « méta-événement », et chaque prestation un événement à part entière. Les différents événements correspondant aux prestations, sont agrégés au « méta-événement » à travers une relation hiérarchique : **sub_event**.

50. <https://www.sirene.fr/sirene/public/static/contenu-base-sirene>

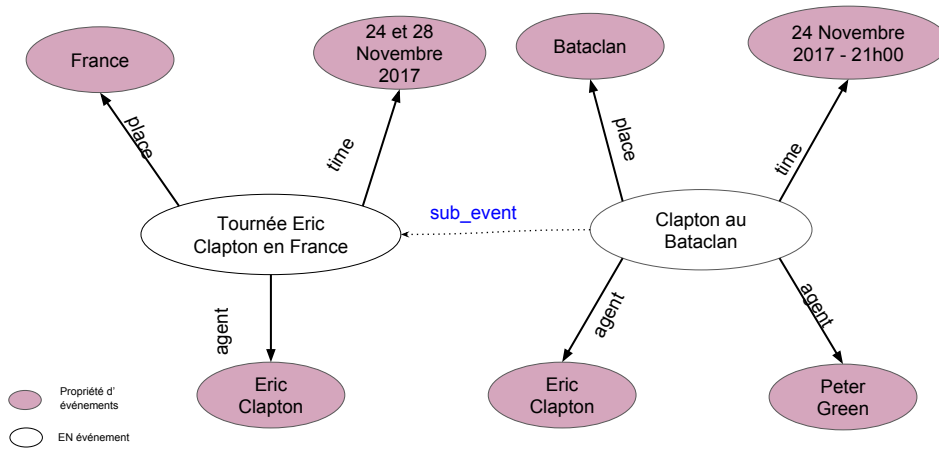


FIGURE 4.1 – Tournée d’Eric Clapton selon l’ontologie EVENT

- **LODE** [176] : c’est une ontologie définie initialement pour la description des événements correspondant à des faits historiques. Elle a évolué avec le temps et elle est utilisée aussi pour la description d’autres types d’événements comme les faits d’actualité ou même les événements socio-culturels. Ce modèle réutilise les classes de l’ontologie DUL⁵¹ pour spécifier les types de propriétés. La figure 4.2 représente les deux prestations de la tournée d’Eric Clapton selon le modèle LODE. Chaque prestation est un événement à part entière, et une relation « **owl:sameAs** » permet d’établir un lien entre événements similaires (prestations d’une même tournée dans ce cas).

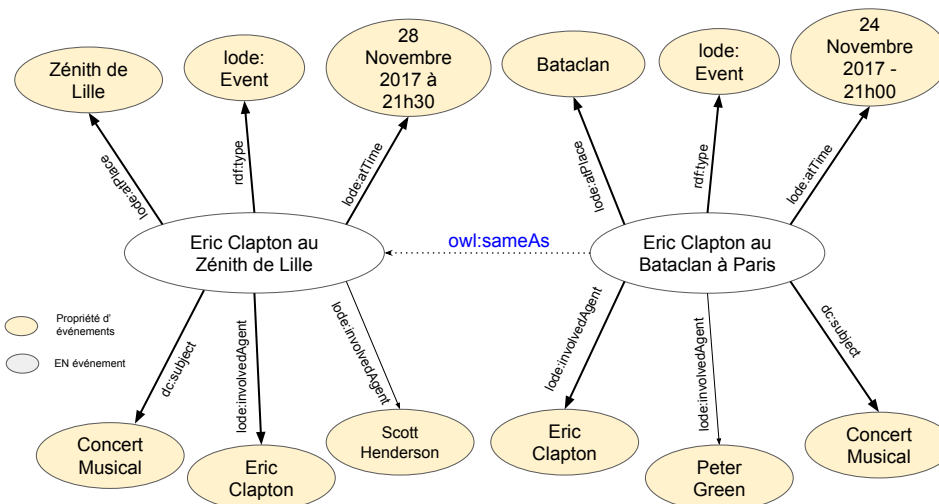


FIGURE 4.2 – Tournée d’Eric Clapton selon l’ontologie LODE

- **SEM** [74] : il s’agit d’une ontologie développée à l’université d’Amsterdam. Elle a une structure très

51. http://ontologydesignpatterns.org/wiki/Ontology:DOLCE+DnS_Ultralite

riche et elle est adaptée pour la représentation des deux types d'événements (faits et événements socio-culturels). Elle permet de spécifier les types des propriétés de l'événement, tout en exprimant des contraintes sur ces propriétés. La figure 4.3 présente deux prestations de la tournée d'Eric Clapton. Comme pour le modèle EVENT, chaque prestation est un événement à part entière, relié à la tournée par une relation dédiée (**sem:hasSubEvent**). Cependant, SEM permet aussi de spécifier des contraintes sur les propriétés et d'associer des catégories aux événements (**sem:eventType**).

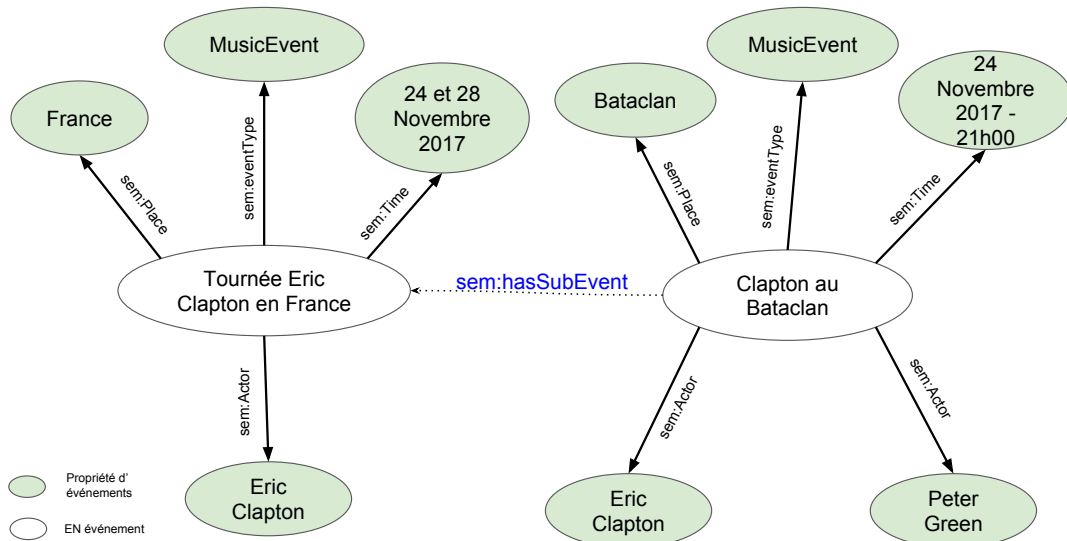


FIGURE 4.3 – Tournée d'Eric Clapton selon l'ontologie SEM

- **h-event**⁵² : c'est une classe du formalisme microformats.org. elle est utilisée pour représenter un événement ainsi que ses propriétés dans une page web. Dans ce modèle, chaque prestation de la tournée est considérée comme un événement à part entière. Cependant, contrairement aux trois modèles de type ontologique présentés précédemment, il n'est pas possible de représenter des relations entre événements. Ci-dessous (listing 4.10) la première prestation de l'événement correspondant à la tournée d'Eric Clapton, représentée dans un page web avec le modèle *h-event*.

```

1 <div class="h-event">
2   <h1 class="p-name">Eric Clapton à Paris</h1>
3   le
4   <h3 class="dt-start">24 Novembre 2017</h3> au
5   <div class="p-location h-card">
6     <h3 class="p-name">Bataclan</h3>
7     ...
8   </div>
9   avec
10  <p class="h-card">
11    <h3 class="p-name">Eric Clapton</h3>
    
```

52. <http://microformats.org/wiki/h-calendar>


```

12     ...
13     </p>
14     et
15     <p class="h-card">
16         <h3 class="p-name">Peter Green</h3>
17         <span class="p-job-title">Guitariste</span>
18     </p>
19 </div>
20

```

Listing 4.10 – Représentation d'une prestation de la tournée d'Eric Clapton avec h-event

- **Event**⁵³ : c'est une classe du formalisme schema.org, qui comme *h-event*, est utilisée pour représenter les événements dans les pages web. Contrairement à *h-event*, elle permet notamment de représenter des relations entre événements. Ce modèle tient compte des différentes catégories d'événements (festivals, événements sportifs, colloques, etc.), qui sont représentées par des classes spécialisées d'*Event* (*BusinessEvent*, *MusicEvent*, etc.). Le listing 4.11 illustre la représentation de la tournée d'Eric Clapton avec ce modèle. Chaque prestation est un événement, de la catégorie « *MusicEvent* » et la relation hiérarchique permettant de les relier à l'événement global est exprimée par la propriété **subEvent**.

```

1     <div itemscope itemtype="http://schema.org/MusicEvent">
2         <h1 itemprop="name">Tournée Eric Clapton en France</h1>
3         Du
4         <span itemprop="startDate" content="2017-11-24">24 Novembre 2017</span>
5         au
6         <span itemprop="endDate" content="2017-11-28">28 Novembre 2017</span>
7         Avec deux prestations :
8         <div itemprop="subEvent" itemscope
9             itemtype="http://schema.org/MusicEvent">
10            <h1 itemprop="name">Eric Clapton à Paris</h1>
11            Au
12            <div itemscope itemtype="http://schema.org/Place">
13                <span itemprop="name">Bataclan</span>
14                ...
15            </div>
16            Sur scène :
17            <div itemprop="performer"
18                itemscope itemtype="http://schema.org/Person">
19                <span itemprop="name">Eric Clapton</span>
20            </div>
21            et
22            <div itemprop="performer" itemscope
23                itemtype="http://schema.org/Person">

```

53. <http://schema.org/Event>

```

24     <span itemprop="name">Peter Green</span>
25   </div>
26 </div>
27
28   <div itemprop="subEvent" itemscope
29     itemtype="http://schema.org/MusicEvent">
30     <h1 itemprop="name">Eric Clapton à Lille</h1>
31   ....
32   </div>
33 </div>
34

```

Listing 4.11 – Représentation de la tournée d’Eric Clapton avec schema.org/Event

- **Le modèle ACE** (Automatic Content Extraction) [132] : c’est le modèle ad-hoc le plus utilisé dans les travaux de recherche pour représenter des événements [164]. Il a été défini dans le cadre de la tâche « Event Detection and Recognition » des campagnes d’évaluation de l’extraction d’événements ACE⁵⁴. Un événement selon ce modèle est défini par plusieurs propriétés notamment un lieu, une date, un type ou des participants, qui, eux même peuvent avoir des rôles (Victim, Buyer, etc.). Ce modèle organise les événements selon 8 types (*Life, Movement, Transaction, etc.*) et 33 sous-types (*Convict, Attack, etc.*). La notion de sous-événement n’est pas prise en compte dans le modèle ACE. De ce fait, pour l’événement lié à la « tournée d’Eric Clapton », toutes les prestations seraient des événements indépendants sans aucun lien entre eux. De plus, d’après Mitamura et al. [120], le modèle ACE est plus adapté pour la représentation des événements de type *fait*. En effet, les types prévus dans ce modèle permettent uniquement de représenter des faits (narrations, actualité, etc.). Aucun type dans ledit modèle ne permettrait de représenter l’événement correspondant à la « tournée de Eric Clapton », qui est de la catégorie « Concert », non définie dans le modèle ACE.

4.3 Bilan

Cette section a permis de présenter les principales catégories de modèles proposées pour représenter les EN. Nous avons également dressé, pour différents types d’EN, une liste non exhaustive de modèles utilisés pour les représenter. Le tableau 4.3 présente la synthèse des modèles décrits dans cette section. Pour chacun d’entre eux, nous spécifions sa catégorie ainsi que le(s) type(s) d’EN ciblé(s).

En ce qui concerne la représentation des EN de type *adresse*, de tous les modèles explorés, c’est le modèle d’Adresse de l’Etalab qui permet de représenter la plus grande variété de propriétés d’adresses. En effet, avec ce modèle, il est possible de représenter une adresse allant du numéro de voie jusqu’au nom de la commune, en passant par les informations administratives (code INSEE, nom AFNOR ...), ou encore les coordonnées géographiques. Cependant, il est davantage focalisé sur l’aspect spatial de l’adresse, ainsi, les informations postales (boîte postale, par exemple) ne sont pas prises en compte. Pour ce qui est des modèles de type standard du web, ils permettent de tenir compte à la fois des aspects géographique et postal d’une adresse. Un modèle comme *h-adr*, par exemple, permet de représenter le numéro de boîte

54. <https://www ldc.upenn.edu/collaborations/past-projects/ace>

postale associé à une adresse (propriété *p-post-office-box*). Ce modèle possède aussi un champ libre (*p-extended-address*) qui permet de spécifier des informations complémentaires pour une adresse comme, le nom d'un bâtiment ou encore le nom d'un lieu-dit. Toutefois, ce modèle offre moins de propriétés que le modèle d'Adresse de l'Etalab.

Concernant la représentation des entreprises, les modèles de la catégorie standard du web sont assez génériques. Ils considèrent les entreprises comme des POI « classiques », et ils se focalisent beaucoup plus sur l'aspect spatial de celles-ci. L'aspect administratif des entreprises (numéro d'immatriculation, capital social, statut juridique, etc.) est complètement ignoré. Le modèle SIRENE, par contre, est plus centré sur l'aspect administratif des entreprises. Toutefois, il ne permet pas d'associer plusieurs secteurs d'activité à une entreprise, ce qui est généralement le cas dans la pratique.

En ce qui concerne les modèles de représentation des événements, tous ceux que nous avons explorés représentent les événements comme des objets à plat, avec toutes les propriétés au même niveau. De ce fait, pour représenter les événements comme des tournées qui peuvent s'étaler sur plusieurs mois et plusieurs lieux distincts, ces modèles considèrent chaque prestation de la tournée comme un événement à part entière, parfois au même titre que la tournée elle-même. Ce qui entraîne des redondances lorsqu'on traite des événements socio-culturels présentant cette spécificité (tournées, tournois, conférences, etc.).

TABLE 4.3 – Quelques modèles de représentation des EN

Modèles	Catégories de modèles				EN ciblées				Références	
	Standard	Web	Ontologies	Ad-hoc	Temps	Lieu	Entité sociale	Entreprise		Événement
TIMEX3	✓				✓					
h-geo	✓					✓				
h-adr	✓					✓				
schema.org/Place	✓					✓				
GeoRSS-Simple			✓			✓				[71]
Adresse (Etablab)				✓		✓		✓		
h-card	✓						✓			
schema.org/Person	✓						✓			
schema.org/Organization	✓						✓	✓		
foaf:Person			✓				✓			
foaf:Organization			✓				✓	✓		
schema.org/LocalBusiness	✓						✓	✓		
SIRENE				✓				✓		
EVENT			✓						✓	[143]
LODE			✓						✓	[176]
SEM			✓						✓	[74]
h-event	✓								✓	
schema.org/Event	✓								✓	
Evenement ACE				✓					✓	[132]

Chapitre 5

Extraction d'entités nommées sur le Web

Deux grandes techniques existent pour l'extraction d'EN sur le web. La première consiste simplement à interroger les bases de données des sites contenant ces EN, via des API⁵⁵ dédiés. Dans ce cas, il faut que ces sites autorisent le partage de leurs informations et mettent à disposition des API pour permettre l'accès aux données. Les travaux de Khrouf et al. [90], par exemple, portent sur la collecte des informations d'événements à partir de plusieurs plateformes du web, en utilisant des API. Les plateformes ciblées sont notamment Eventful.com⁵⁶, Last.fm⁵⁷, Upcoming⁵⁸ et flickr⁵⁹. Cependant, l'accès à ces API est généralement payant, et quand bien même il est libre, le nombre de requêtes est limité. Une deuxième technique consiste à analyser le contenu du web pour filtrer un sous-ensemble précis dont les pages seront analysées pour la tâche d'extraction d'EN. Cette technique permet notamment de lever le verrou des quotas et des accès payants. Dans le cadre de notre travail, nous nous intéressons à cette deuxième technique.

En ce qui concerne l'extraction d'EN à partir de textes provenant du web, deux stratégies sont généralement utilisées :

- la première exploite les méta-données (respectant les formalismes de partage d'informations sur le web : microformats.org ou schema.org principalement) embarquées dans les balises de pages, pour extraire les EN qu'elles contiennent. Cependant, très peu de sites web utilisent ces formalismes pour publier les informations ;
- la deuxième stratégie consiste à extraire le texte des pages en éliminant notamment le balisage HTML, avant d'appliquer des techniques d'extraction d'EN sur du texte.

Pour notre problématique, nous nous focalisons sur cette deuxième stratégie, car elle permet de traiter le maximum de pages web.

Ce chapitre s'organise en trois sections. La première est dédiée aux prétraitements permettant de passer du web à une collection de textes ; la deuxième sous-section présente les principales approches d'extraction d'EN dans le texte ; et la dernière sous-section cible des travaux dédiés à l'extraction d'EN

55. Interface permettant la communication entre applications : https://fr.wikipedia.org/wiki/Interface_de_programmation

56. API Eventful : <https://api.eventful.com>

57. API Last.fm : <https://www.last.fm/fr/api>

58. API Upcoming : <https://www.programmableweb.com/api/upcomingorg>

59. API flickr : <https://www.flickr.com/services/api/>

de différents types.

5.1 Du web au texte

L'extraction d'un type d'EN donné ne nécessite pas l'analyse de tout le contenu du web. En effet, lancer un traitement à l'échelle de tout le web serait très gourmand en ressources et en temps, d'où l'idée de filtrer le contenu du web pour cibler l'analyse sur un ensemble de pages ou de sites uniquement. L'opération de filtrage du web et la transformation du contenu des pages en texte constituent les prétraitements. Nous en présentons trois dans cette section : le filtrage du web, la classification des pages web et l'identification des blocs structurels dans une page web. Selon le corpus traité, ces trois prétraitements peuvent être combinés pour la construction de la collection de textes à analyser pour extraire des EN.

5.1.1 Filtrage du web

C'est l'opération qui permet de sélectionner un sous-ensemble de sites web ou de pages, traitant d'un sujet précis. Ce prétraitement est plus connu sous le nom de « *web crawling* ». C'est un thème de recherche en plein essor au regard du très grand volume de pages contenues sur le web. Trois principales approches existent pour réaliser cette opération : l'approche « *focused crawling* » [43], l'approche « *semantic crawling* » [137] et l'approche « *semantic web crawling* ». Les deux premières exploitent les hyperliens entre pages pour parcourir le web, alors que la troisième s'appuie sur le web de données.

5.1.1.1 Approche « *focused crawling* »

Elle exploite les hyperliens entre pages et s'appuie sur l'hypothèse selon laquelle, les pages traitant de sujets similaires s'inter-référencent. En entrée du processus, nous avons un sujet (ensemble de mots clés par exemple) et une page web donnée (que l'on appelle page initiale). Toutes les pages référencées via la page initiale sont analysées, et des scores de pertinence avec le sujet sont calculés. Les pages ayant un score au dessus d'un seuil sont sélectionnées et le processus est répété sur chacune de ces nouvelles pages, jusqu'à atteindre un quota ou jusqu'à ce qu'il n'y ait plus de pages à analyser. Le calcul de pertinence entre page web et sujet est généralement évalué en se basant sur les mesures de pertinence entre requêtes et documents, utilisées en recherche d'information (le modèle vectoriel, par exemple [157]). Certains travaux comme ceux de Rae et Murdock [141] mettent en oeuvre un tel processus pour l'identification du sous-ensemble du web traitant des POI. Ils utilisent la page d'accueil du service de recherche de Bing⁶⁰ comme page initiale. Le sujet est défini par les mots clés de la requête.

5.1.1.2 Approche « *semantic crawling* »

Elle est très proche de la première approche, et comme celle-ci, elle exploite les hyperliens entre pages. En entrée du processus, nous avons une ressource sémantique, un sujet défini à partir des concepts de la ressource et une page initiale. À la différence de l'approche par « *focused crawling* », le score de pertinence entre le sujet et une page est évalué en se basant sur la proximité sémantique entre concepts caractérisant le sujet et ceux identifiés dans la page. La section 6.1.2 détaille les principales approches de calcul de proximité sémantiques entre concepts.

60. <https://www.bing.com/>

5.1.1.3 Approche « semantic web crawling »

Contrairement aux deux approches précédentes, les liens entre pages ne sont pas exploités ici pour le parcours du web. En entrée du processus, nous avons une ressource du web sémantique et un sujet défini à partir des concepts de cette ressource. Dans une telle ressource, les concepts ont des propriétés qui recensent les pages faisant référence à ceux-ci. Pour les ressource en RDF, par exemple, les propriétés « *rdfs::isDefinedBy* » et « *rdfs::seeAlso* » répertorient des pages web qui traitent des concepts. Le filtre parcourt donc la ressource sémantique concept par concept pour identifier toutes les pages faisant référence au sujet. Slug [144] est un exemple de *crawler* exploitant cette approche. Les robots du moteur de recherche sémantique Swoogle [44] également utilisent cette technique de filtrage pour la construction de leurs index.

5.1.2 Classification de pages Web

C'est l'opération qui permet de regrouper les pages d'une collection en ensembles selon certaines caractéristiques. Cette opération est plus connue sous son nom en anglais : « *webpage classification* ». À la différence du « *web crawling* » dont le but est d'identifier les sites ou pages web relatifs à un sujet donné, l'enjeu ici est de regrouper des pages web, selon des critères précis. D'après Choi et al. [30], on peut classer les pages web selon :

- le thème traité : ainsi, les pages d'un site d'e-commerce peuvent être regroupées selon les différentes catégories de produits mis en vente sur le site ;
- la fonction de la page : pour le site d'une entreprise par exemple, on peut faire la distinction entre la page d'accueil, les pages de catalogue de produits et les pages de contacts.

Les techniques utilisées pour la classification de pages web sont généralement basées sur de l'apprentissage automatique. Selon les propriétés de la page utilisées pour la classification, nous avons deux grandes approches. La première exploite uniquement le texte des pages et la deuxième, en plus du texte, exploite également la structure des pages et les médias qu'elles contiennent.

5.1.2.1 Approches exploitant uniquement le texte des pages

Les critères de classification dans ce cas sont les mots du texte, les mots « vides » [78] étant éliminés au préalable. Ces approches sont adaptées pour la classification selon le thème [30]. Kriegel et al. [93], par exemple, utilisent l'algorithme des *k* plus proches voisins [125], pour la classification des pages de yahoo!⁶¹ selon les différentes catégories du site (sport, économie, etc.). Nous avons également les travaux de Largillier et al. [96] qui exploitent une approche par apprentissage, basée sur les forêts d'arbres de décisions [150] pour identifier les pages web avec du contenu pour adultes.

5.1.2.2 Approches exploitant texte, structure et médias des pages

Dans ce cas, les autres informations contenues dans la page web notamment les balises de structuration ou même les images, viennent compléter le texte pour la construction du classifieur. Ces approches sont adaptées aussi bien pour la classification selon le thème, que pour la classification selon la fonction [30]. Kalva et al. [88], par exemple, combinent les images et le texte pour classifier une collection de pages web. Les images sont classifiées en utilisant les réseaux de neurones [189] et le texte l'est en utilisant un réseau

61. <https://fr.yahoo.com>

bayésien [77]. De même, Meshkizadeh et al. [118] utilisent en plus de la structure, les liens entre pages et le texte de celles-ci, pour construire un classifieur bayésien selon les catégories de l'annuaire ODP⁶².

5.1.3 Identification des blocs structurels d'une page

C'est l'opération qui permet, dans une page web, d'identifier les différents blocs structurels afin d'en sélectionner ou d'en éliminer certains. Un bloc structurel (ou bloc visuel [51]) peut correspondre, par exemple, au menu d'un site, ou au bloc principal dans un article de presse (celui qui contient le corps de l'article). La figure 5.1, extraite de la thèse de Faessel [51], illustre un découpage en blocs structurels d'une page web pour des niveaux de granularité différents. L'objectif de ce prétraitement est de centrer le processus d'extraction d'EN sur les blocs « susceptibles » d'en contenir.



FIGURE 5.1 – Découpage en blocs structurels d'une page web, extraite de la thèse de Faessel [51]

Pour identifier les différents blocs structurels d'une page web, deux principales approches sont utilisées : la première est basée sur des heuristiques et la deuxième s'appuie sur de l'apprentissage automatique. Ces deux approches exploitent la structure en arborescence du graphe représentant le DOM⁶³ des pages.

5.1.3.1 Approches d'identification de blocs structurels basées sur des heuristiques

Dans cette catégorie d'approches, des patrons personnalisés sont construits pour analyser le DOM des pages et identifier les blocs. Liu et al. [110] par exemple, proposent une approche qui repose sur l'hypothèse selon laquelle, un bloc dans une page web doit contenir un certain nombre de nœuds au moins, et ceux-ci doivent être au même niveau de la hiérarchie du graphe. De même, Cai et al. [25],

62. Open Directory Projet : ancien nom de DMOZ (<http://www.dmoz.org/>), annuaire de sites web classés par catégories de 1998 à 2017

63. Document Object Model : modèle à partir duquel sont construites les pages web conformément aux recommandations du W3C

considèrent que les blocs visuels d'une page sont mis en évidence par des balises particulières (`<hr/>`, `<table>`, etc.).

5.1.3.2 Approches d'identification de blocs structurels basées sur de l'apprentissage

Pour cette catégorie d'approches, un modèle d'apprentissage est utilisé pour identifier les blocs. Thamyet et al. [174], par exemple, utilisent une technique par apprentissage supervisé pour identifier les blocs structurels d'une page. Leur algorithme d'apprentissage exploite à cet effet les structures répétitives de la page. Kohlschutter et al. [92] utilisent, eux aussi, une approche par apprentissage supervisé pour l'identification des blocs dans des pages relatives à des articles de presse. Leur algorithme est basé sur la concentration de texte et leur objectif est d'éliminer les blocs « génériques » (boilerplates), c'est à dire ceux qui ne sont pas porteurs d'information.

Nous rappelons que selon le contexte, un seul ou une combinaison de ces pré-traitements peut être exploité(e) pour constituer la collection de pages web à analyser. Le texte obtenu en éliminant les balises HTML peut ensuite être traité en appliquant des approches d'extraction d'EN dédiées aux textes non ou peu structurés.

5.2 Approches d'extraction d'entités nommées dans le texte

L'extraction d'information peut être définie comme une tâche dont le but est de construire des informations structurées à partir de corpus de documents non ou peu structurés. L'extraction d'EN est une sous-tâche de celle-ci, et sa mise en œuvre dans des textes non ou peu structurés posent plusieurs problèmes. Nous avons notamment :

- **la détection de syntagmes correspondant aux EN parmi tous les mots du texte** : ce problème découle du fait que les EN sont généralement noyées dans le texte sans marquage particulier. Considérons la phrase : « *Ngolo Kanté* est le meilleur milieu de terrain de sa génération ». « *Ngolo Kanté* » ici est une EN de type *personne*, noyée dans la phrase et l'enjeu est de l'identifier correctement. L'état de l'art de Nadeau et al. [126] présente la synthèse des principales techniques proposées pour ce problème ;
- **la sur-composition** : c'est le cas où un syntagme correspondant à une EN, contient d'autres syntagmes faisant référence à d'autres EN ([61], [122]). Considérons le POI associé à l'« *Université Paul Sabatier de Toulouse* ». Le syntagme « *Paul Sabatier* » correspond à une EN de type *personne*, alors que le terme « *Toulouse* » fait référence à la ville de Toulouse dans le sud de la France. La difficulté dans ce cas est d'extraire l'EN dans sa globalité en tenant compte de cette sur-composition ;
- **l'agrégation des propriétés d'EN** : ce problème est spécifique à l'extraction d'EN complexes, dont les différentes propriétés sont éparpillées dans le texte. Généralement l'approche adoptée pour ce problème consiste à identifier les différentes propriétés séparément, puis d'agréger ces annotations pour reconstruire l'EN. Le principal problème ici est l'agrégation correcte de ces annotations, en tenant compte du fait qu'un même texte peut contenir plusieurs instances d'EN complexes.
- **l'ambiguïté contextuelle** : ce problème se pose lorsque qu'un même syntagme fait référence à deux EN distinctes. A titre illustratif, le terme « *Paris* » peut faire référence à la ville de Paris capitale française, ou à la ville de Paris au Texas (USA) : on parle alors d'ambiguïté Géo/Géo.

De même, le terme « *Paris* » peut faire référence dans un contexte précis à Paris le lieu (USA ou France) ou à la personne « Paris Hilton » : il s'agit d'une ambiguïté Geo/Non-Géo. Plusieurs travaux, parmi lesquels ceux de Bensalem et al. [13] ou Buscaldi et al. [24], proposent des techniques pour résoudre ces ambiguïtés dans le cas des EN spatiales.

Dans cette section, nous nous intéresserons aux approches proposées pour extraire les EN dans le texte. Nous organisons ces approches en quatre catégories : les approches basées sur les patrons, les approches par apprentissage, les approches exploitant les ressources externes et enfin les approches hybrides.

5.2.1 Approches par patrons

Les approches de cette catégorie sont dites inductives. En effet, l'observation d'un échantillon de textes, permet d'identifier des formes régulières [39] utilisées pour exprimer les EN ciblées. Ces formes sont traduites en patrons, qui sont généralement exprimés en exploitant les propriétés morpho-syntaxiques des termes du texte, ainsi que des mots clés [17]. L'extraction d'information une fois les patrons construits, se fait généralement en plusieurs étapes : (i) une première étape a pour but de réaliser l'étiquetage morpho-syntaxique des mots du texte ; (ii) une deuxième étape intervient par la suite pour identifier les mots clés dans le texte ; (iii) et enfin une dernière étape analyse chaque unité de texte (la phrase par exemple) à partir des mots clés, pour identifier les fragments qui « déclenchent » les patrons correspondants. Supposons par exemple, qu'on s'intéresse à l'extraction des noms d'inventeurs dans du texte. Le patron ci-dessous permettrait d'identifier « Graham Bell » dans la phrase suivante : « *Graham Bell a inventé le téléphone* ».

$$\underbrace{\text{Nom_Propre}}_{\text{Inventeur}} \text{ a inventé } \text{Article} \text{ } \text{Objet}$$

Dans ce patron, un nom d'inventeur potentiel est un syntagme correspondant à un nom propre, suivi par l'expression « a inventé », puis d'un article et d'un nom d'objet. Un lexique ou dictionnaire de mots clés contenant des noms d'objets (téléphone, ordinateur, papier, etc.) est utilisé pour induire le déclenchement du patron. Plusieurs travaux exploitent cette approche pour l'extraction d'information dans différents contextes ([50], [4], [21], [107]). Toutefois, l'efficacité des patrons ne peut être garantie que lorsque les formes d'expressions des EN sont dénombrables et susceptibles d'être explicitement définies : on parle alors de formes régulières. Dans le cas contraire, le silence du processus d'extraction serait très élevé, parce que plusieurs EN pourtant valides, n'auraient pas été extraites [26].

5.2.2 Approches par apprentissage

Pour cette catégorie d'approches, les formes d'expression d'une EN sont déduites en exploitant un échantillon du corpus à analyser. Les règles permettant de déduire ces formes sont inférées à partir de méthodes statistiques et stockées dans des modèles d'apprentissage : les approches de cette catégorie sont dites déductives. Les techniques de traitement automatique du langage (TAL) et les algorithmes d'apprentissage sont utilisés pour la construction du modèle d'inférence. Il existe deux familles de techniques d'apprentissage utilisées pour extraire les EN dans le texte : les techniques par apprentissage supervisé [178] et les techniques par apprentissage non supervisé [102].

En ce qui concerne les techniques basées sur de l'apprentissage supervisé, les EN ciblées sont étiquetées par un expert sur un échantillon de textes représentatif du corpus : c'est le jeu d'entraînement. Des algorithmes dédiés sont mis en œuvre sur ce jeu d'entraînement pour inférer le modèle qui permettra de

détecter des EN sur un corpus plus large. Plusieurs algorithmes peuvent être utilisés à cet effet. Nous avons notamment les chaînes de Markov cachées [14], les arbres de décisions [163], les modèles basés sur l'évaluation de l'entropie maximale [22] ou encore le Conditional Random Fields (CRF) [116].

Pour les techniques par apprentissage non supervisé, aucun étiquetage n'est nécessaire au préalable. Des méthodes statistiques permettent d'analyser l'ensemble du corpus et d'inférer le modèle de détection d'EN en exploitant les ressemblances dans ce corpus. En effet, ces méthodes vont regrouper les syntagmes du texte selon les propriétés communes qu'ils présentent. Nous pouvons avoir comme propriétés notamment, la classe morpho-syntaxique, la forme des mots (capitalisation, majuscule, minuscule) ou même les séquences de mots. Shinyama et al. [169] proposent, par exemple, une approche non supervisée permettant d'extraire des EN dans le texte en exploitant la distribution des mots sur le corpus. Une condition nécessaire pour garantir l'efficacité des approches d'extraction basées sur l'apprentissage non supervisé, est le volume important du corpus traité [49].

Les approches par apprentissage présentent certaines limites. En effet, elles requièrent parfois, un travail laborieux d'annotation manuelle pour la constitution du jeu d'entraînement dans le cas de l'apprentissage supervisé. De même, le choix des paramètres permettant d'obtenir un modèle d'apprentissage optimal constitue un véritable verrou. Enfin, parce que l'apprentissage n'est pas une méthode déterministe, il est très difficile d'expliquer certains dysfonctionnements du processus d'extraction d'information, surtout pour les techniques par apprentissage non supervisé. Cependant, la force des approches par apprentissage est qu'elles permettent une identification efficace des EN, quand bien même leurs formes d'expressions ne sont pas régulières.

5.2.3 Approches basées sur les ressources externes

Pour cette catégorie d'approches, des ressources de type connaissance, décrivant un domaine précis sont projetées sur le texte pour identifier les EN. Ces ressources peuvent être des ontologies ou des thésaurus. Dans les deux cas, elles sont décrites par un vocabulaire spécifique qui sert à annoter le texte. Les récents développements dans le domaine du web sémantique ont fait des ontologies, l'une des ressources les plus utilisées pour l'extraction des EN dans le texte [65]. Soner et al. [170], par exemple, utilisent une ontologie pour extraire des EN propres au domaine du football.

Dans la pratique, la projection de la ressource sur le texte pour l'identification des EN se fait en plusieurs étapes [37]. Chaque syntagme du vocabulaire de la ressource est d'abord découpé en unité textuelle (tokenisation). À chaque unité textuelle est assignée une information morphologique : le lemme. Les mots du texte sont traités de la même façon. Après avoir tokenisé et lemmatisé la ressource et le texte, ce dernier est parcouru pour repérer les syntagmes lemmatisés de la ressource. Tous les syntagmes identifiés dans le texte sont annotés du concept ou de la relation qu'ils caractérisent.

Reprenons la phrase : « *Graham Bell a inventé le téléphone* ». Dans la ressource DBpedia, « Graham Bell » est un des noms utilisé pour désigner la personne « Alexander Graham Bell », qui y est référencé comme un « inventeur ». De ce fait, cette ressource peut être utilisée pour annoter le syntagme « Graham Bell » de l'étiquette « Inventeur » dans la phrase.

La principale limite des approches basées sur les ressources externes est l'ambiguïté contextuelle. En effet, la projection de la ressource ne tient pas compte du contexte dans lequel les syntagmes apparaissent dans le texte. Des nombreux travaux ([36]) proposent des techniques et stratégies pour lever certaines ambiguïtés. De même, les ressources sont parfois incomplètes pour décrire la connaissance d'un domaine.

Soit parce que le domaine décrit a évolué et que les nouveaux changements n'ont pas encore été intégrés, soit simplement parce que le vocabulaire de la ressource est pauvre pour décrire le domaine. C'est pour résoudre ce dernier problème qu'à émergé le domaine de l'enrichissement lexical des ressources de type connaissance [146]. Son objectif principal étant d'enrichir le vocabulaire de ces ressources, en utilisant principalement des approches par apprentissage.

5.2.4 Approches hybrides

De nombreux travaux proposent des techniques d'extraction d'EN, qui combinent les approches précédentes. L'objectif visé est de compenser les limites des unes par les forces des autres [41]. Les approches qui s'inscrivent dans cette logique sont appelées approches hybrides. En ce qui concerne les approches combinant patrons et apprentissage, les patrons permettent d'améliorer le processus de construction du modèle d'apprentissage, alors que l'apprentissage aide à la découverte de nouveaux patrons. On appelle la technique combinant ces deux approches l'apprentissage de patrons ([148], [27]) . De même, le déclenchement de certains patrons est parfois induit par l'identification de certains mots clés. Ces mots clés peuvent être projetés sur une ressource de type connaissance pour améliorer les patrons. Considérons par exemple la phrase : « *Graham Bell a conçu le téléphone* ». En utilisant la ressource WordNet [119], on se rend compte que les verbes « concevoir » et « inventer » sont synonymes. Par conséquent, le patron écrit en 5.2.1 reste valide pour l'annotation de « Graham Bell », avec l'étiquette « Inventeur ». Enfin, les ressources peuvent être utilisées pour donner de la connaissance à priori à un système d'apprentissage.

5.3 Travaux dédiés à l'extraction d'EN

Plusieurs travaux mettent en œuvre les approches d'extraction présentées dans la section précédente pour l'identification d'EN dans le texte. Les chaînes d'extraction résultantes sont habituellement mises en œuvre via des plateformes comme GATE ANNIE⁶⁴, Stanford NER⁶⁵, OpenNLP⁶⁶ ou encore LinguaStream [15]. Dans cette section, nous explorons des travaux dédiés respectivement à l'extraction d'EN temporelles, spatiales, sociales, entreprises et événements.

5.3.1 Extraction des EN temporelles

Martins et al. [115] proposent une chaîne d'extraction d'EN temporelles dans le texte. Ces EN sont de deux sous-types : les périodes historiques (la révolution française, l'antiquité, etc.) et les dates (Septembre 2001, 17 Mars 1995, etc.). L'extraction des périodes historiques est réalisée en exploitant une ressource dédiée, alors que les dates sont extraites avec des patrons sous forme d'expressions régulières. Strotgen et Gertz [171] ont développé un système très connu pour l'identification d'EN temporelles dans du texte : Heideltime⁶⁷. Ce système a la particularité de traiter les textes de plusieurs langues, notamment l'anglais, l'allemand, l'arabe, le français. L'approche d'extraction mise en œuvre dans le système Heideltime est basée sur les patrons. Le Parc-Lacayrelle et al. [98] ou encore Alonso et al. [6] exploitent aussi une approche par patrons pour extraire des EN temporelles dans le texte. Ces patrons sont construits à partir des propriétés morpho-syntaxiques des mots du texte.

64. <https://gate.ac.uk/ie/annie.html>

65. <https://nlp.stanford.edu/software/CRF-NER.shtml>

66. <https://opennlp.apache.org/>

67. <https://code.google.com/archive/p/heideltime/>

D'autres travaux exploitent des techniques d'apprentissage automatique pour l'extraction des EN temporelles dans le texte. C'est le cas, par exemple, de ceux de Filannino et al. [54], qui utilisent l'algorithme CRF (conditional Random Fields) [116], pour construire le système *ManTIME*. Ce système cible l'extraction de différentes formes d'expressions temporelles dans le texte (dates, périodes, etc.).

5.3.2 Extraction des EN spatiales

De nombreux travaux s'intéressent à l'extraction des EN de type lieu, nous avons, par exemple, les travaux de Vaid et al. [177] ou ceux de Wang et al. [183] qui exploitent des ressources de lieux pour extraire ce type d'EN dans le texte. De même, les travaux de Nguyen et al. [129] permettent d'extraire les relations entre lieux dans le texte. Cette proposition consiste à représenter la phrase sous forme de graphe, puis d'exploiter l'analyse morpho-syntaxique de chaque phrase pour en déduire, à travers des patrons, les relations entre lieux.

D'autres travaux plus spécifiques sont dédiés à l'extraction d'adresses dans le texte. Nous les organisons selon les quatre catégories d'approches d'extraction d'EN énumérées dans la sous-section précédente.

- Dans la catégorie des approches par apprentissage, Berenike et al. [111], par exemple, utilisent l'algorithme Conditional Random Fields (CRF) [116] pour la reconnaissance d'adresses. Taghva et al. [87], quant à eux, utilisent un algorithme basé sur les chaînes de Markov cachées [190].
- Dans la catégorie des approches basées sur les patrons, Nesi et al. [128], utilisent des patrons basés sur l'analyse morpho-syntaxique du texte pour extraire des adresses italiennes. La particularité de leur contexte de travail est qu'ils font l'hypothèse que les adresses à extraire suivent des standards d'écriture. Ce qui n'est pas toujours vrai en pratique et particulièrement sur le web. Li et al. [106] proposent également une chaîne d'extraction d'adresses de services d'urgence aux États-Unis. Leur contexte de travail est similaire à celui de Nesi et al., et leur processus d'extraction est basé sur des expressions régulières. Ahlers et al. [3] proposent également une approche basée sur des patrons pour l'extraction et la validation d'adresses allemandes. Leur proposition permet d'extraire un grand nombre de formes d'adresses, en exploitant plusieurs dictionnaires, dont un qui répertorie tous les noms de voies en Allemagne.
- Dans la catégorie des approches exploitant des ressources externes de type connaissance, Borges et al. [19] proposent un système d'extraction et de géocodage d'adresses brésiliennes. Leur processus d'extraction est basé sur une ontologie **OnLocus** [20]. Cette ontologie décrit la structure des adresses et des gazetiers sont exploités pour identifier les différentes propriétés.
- Enfin, la catégorie des approches hybrides est illustrée par les travaux de Yu [188]. L'objectif est de comparer plusieurs systèmes d'extraction d'adresses américaines sur un corpus de sites web d'entreprises et d'institutions. Trois systèmes hybrides de détection d'adresses sont expérimentés dans ces travaux : (i) le premier combine patrons et apprentissage ; (ii) le deuxième combine apprentissage et ressources externes ; (iii) et le troisième combine patrons, apprentissage et ressources externes. Le premier et le troisième système permettent d'obtenir les meilleures performances sur leur corpus. En effet, pour le système (ii), l'exploitation des ressources est parfois source d'ambiguïté et donc d'annotations incorrectes. Par ailleurs, Schmidt et al. [162] exploitent également une approche hybride pour l'extraction d'adresses d'entreprises dans le texte de leurs sites web. Leur proposition exploite des ressources et des patrons pour identifier les propriétés d'adresse, ainsi que des informations complémentaires, notamment le nom de l'entreprise. Les adresses traitées suivent

des standards d’écriture et les noms d’entreprises sont toujours placé en début d’adresse, dans leur contexte.

5.3.3 Extraction des EN sociales

En ce qui concerne l’extraction d’EN *sociales* dans le texte, Shaalan et al. [166] proposent un système nommé « *PERA* » pour la reconnaissance automatique des noms de personnes dans des textes en arabe en tenant compte de la sur-composition de ceux-ci (couples nom-prénom, noms composés, etc.). Ce système est basé sur des patrons et exploite des ressources de noms de personnes disponibles sur le web. Le système *UQAM-NTL* développé par Le et al. [99], quant à lui, est dédié à l’extraction d’EN *sociales* dans les tweets. Parmi les entités extraites, il y a des noms de personnes, des noms d’entreprises, des noms d’artistes, des noms d’établissements (hôpitaux, stations services, etc.) et des noms d’équipes sportives. Le système *UQAM-NTL* est basé sur une approche par apprentissage supervisé et c’est l’algorithme CRF [116] qui est mis en œuvre.

5.3.4 Extraction d’EN entreprise

Très peu de travaux ont porté sur l’extraction d’entités entreprises dans le texte. En effet, ceux qui s’y sont intéressés les traitent comme des POI élémentaires (restaurants, commerces, établissements, etc.). C’est le cas par exemple de Rae et al. [141], ou de Corradi et al. [34], qui se sont focalisés sur l’extraction de POI dans des documents textuels. L’enjeu dans tous ces travaux est uniquement d’extraire les propriétés de base de l’entreprise, à savoir son nom, sa localisation et parfois les contacts.

Les travaux d’Alhers ([2] et [3]) vont plus loin et proposent une chaîne « ad-hoc » pour extraire des informations métiers des entreprises, avec des traitements dédiés. L’objectif visé est d’enrichir et de consolider la base de données des Pages Jaunes allemandes. Le corpus utilisé à cet effet est l’ensemble des sites web d’entreprises référencées dans l’annuaire DMOZ. Les informations extraites sont les adresses, les informations de contact (numéros de téléphone, emails, etc.), les informations commerciales (les marques et noms de produits) et les catégories d’activité en exploitant une ressource construite à partir du vocabulaire propriétaire des Pages Jaunes.

5.3.5 Extraction d’EN événement

L’extraction des événements dans le texte dépend de leurs types (faits ou événements socio-culturels), mais aussi de la taille des textes analysés. Pour ce qui est de la taille des textes analysés, on peut avoir des textes courts provenant généralement de posts issus des réseaux sociaux (quelques mots), ou des textes longs (article de presse, texte d’une page web).

5.3.5.1 Extraction d’EN événement dans les textes courts

En ce qui concerne l’extraction d’EN *événement* dans le texte court, une des difficultés majeures vient du fait que le texte contient beaucoup d’abréviations, ce qui rend l’analyse linguistique difficile. Cependant, le fort contexte découlant de la taille réduite du texte à analyser limite les ambiguïtés. Ainsi la difficulté liée au contexte bruité des textes analysés ne se pose que très peu dans ce cas (à titre de rappel, le contexte bruité découle du fait que, des EN de même type qu’une propriété d’EN complexe peuvent apparaître dans le texte : c’est le cas, par exemple, de la date de tenue d’un événement et de

la date de mise en vente des billets). Généralement, c'est l'apprentissage qui est utilisé pour extraire les événements dans des textes courts. Ritter et al. [149] ou Lin et al. [109], par exemple, proposent une approche par apprentissage supervisé pour l'extraction des faits dans un flux de texte provenant de réseaux sociaux. De même Li et al. [104] utilisent une approche basée sur l'apprentissage supervisé, pour identifier des événements socio-culturels dans le texte de posts des réseaux sociaux.

D'autres travaux comme ceux de Becker et al. [12] exploitent les méta-données embarquées dans les documents multimédias des réseaux sociaux pour découvrir des événements de type socio-culturels. Ces méta-données correspondent généralement à des textes courts (description de la photo, tags, titre ...). Leur approche consiste à segmenter l'ensemble des documents à traiter en clusters⁶⁸, les documents d'un même cluster décrivant probablement le même événement. Dans ces travaux, une intervention humaine est nécessaire pour définir explicitement l'événement caractérisé par les documents d'un même cluster.

5.3.5.2 Extraction d'EN événement dans les textes longs

Plusieurs travaux s'intéressent également à l'extraction des deux types d'événements dans des textes longs. Foley et al. [55], par exemple, proposent une approche par apprentissage basée sur les modèles de langues pour extraire des propriétés d'événements de type socio-culturels dans les pages web. Ces pages peuvent contenir plusieurs événements et ils utilisent donc des algorithmes « dédiés » pour agréger les propriétés annotées. Kuzey et al. [94], quant à eux, utilisent une approche basée sur des patrons pour extraire les événements de type faits historiques référencés dans Wikipedia. Serrano et al. [165] proposent une approche hybride pour l'extraction d'événements de type faits d'actualité dans un corpus de dépêches de presse. Cette approche combine deux annotateurs pour l'identification de certaines propriétés d'événement (temps, lieu et agents). Le premier est basé sur des patrons et le second sur l'apprentissage de patrons. Une ontologie dédiée est utilisée pour associer des catégories aux événements extraits (crash, kidnapping, etc.). Dans un contexte de surveillance épidémique, Lejeune et al. [101] ont proposé un système nommé *DANIEL*⁶⁹, dédié à la détection d'épidémies (événements de type *fait*) déclarées dans les articles de presse. Le point fort de ce système est qu'il permet de traiter le texte indépendamment de la langue utilisée. *DANIEL* exploite une ressource construite à partir de Wikipedia et des patrons pour annoter les noms de maladies et les noms de lieux dans lesquels elles ont été signalées. Les patrons construits s'appliquent non pas au niveau des mots, mais au niveau des caractères, ce qui permet de gérer les différentes déclinaisons de noms de maladies et de noms de lieux suivant différentes langues. Notons le verrou liée à l'extraction d'événements dans un contexte bruité n'est pas traité par tous les travaux.

D'autres travaux ciblent l'extraction d'événements dans les textes longs en tenant compte de l'existence d'un contexte bruité. Orlando et al. [134] proposent, par exemple, un système d'extraction d'événements socio-culturels dans des articles de presse. L'annotation se fait en deux phases. Pour la première phase, des patrons et ressources sont utilisées pour identifier toutes les propriétés d'événements : ce sont des propriétés candidates. Cette phase traite le texte sans tenir compte de l'éventualité d'un contexte bruité. La deuxième phase intervient par la suite pour regrouper les différentes propriétés en EN *événement*. Pour cette phase, le produit cartésien des différents ensembles de propriétés candidates permet de construire une liste d'événements « potentiels » et l'interrogation d'un moteur de recherche (Google) permet de leur associer des scores pour la validation. Enfin, Les k meilleurs sont retenus comme étant des événements « réels ». Supposons, par exemple, qu'on représente un événement par un artiste, un lieu et une date.

68. Un cluster est une grappe d'objets ayant des propriétés similaires

69. Data Analysis for Information Extraction in any Language : <https://daniel.greyc.fr/public/index.php?id=1577>

On considère par ailleurs, qu'on s'intéresse à l'extraction d'événements dans l'extrait de texte suivant : « *Eric Clapton se produira le 24 Novembre 2017 au Bataclan à Paris, l'ouverture des ventes de billets est prévue pour le 16 Juin 2017 (...)* ». La première phase de l'approche d'Orlando et al. donnerait les deux événements potentiels suivants : $e_1 = \langle \text{Eric Clapton, 24-11-2017, Bataclan Paris} \rangle$ et $e_2 = \langle \text{Eric Clapton, 16-06-2017, Bataclan Paris} \rangle$. Sur les sites de billetterie, les promoteurs ont tendance à renseigner dans la description de leur événement, la date de mise en vente de tickets. De ce fait, les deux événements « potentiel » ici auraient des scores élevés et pourtant seul le premier est un événement « réel ».

5.4 Bilan

Cette section a présenté les différentes approches utilisées pour extraire des EN sur le web, en exploitant le contenu textuel des pages. Certains prétraitements (filtrage du web, classification de pages web, etc.) peuvent être mis en œuvre pour passer du web à une collection d'extraits de textes. Cette collection est ensuite analysée en utilisant des approches basées sur des patrons, ou bien des approches par apprentissage, ou encore des approches exploitant des ressources externes, ou même une combinaison de celles-ci.

Le tableau 5.1 présente la synthèse des travaux dédiés à l'extraction d'EN dans le texte. Pour chacun d'entre eux, le type d'EN ciblé et l'approche d'extraction mise en œuvre est spécifiée. Pour ce qui est des EN temporelles et de type lieu, ce sont les approches basées sur des patrons et des ressources qui sont les plus utilisées. Ceci se justifie par le fait que les formes d'expressions de ces EN sont généralement régulières. De plus, pour les EN de type lieu, certaines sont décrites dans des ressources dédiées. En ce qui concerne l'extraction d'adresses, les approches par patrons sont les plus utilisées au regard des formes régulières qu'elles suivent. Cependant l'identification de certaines propriétés nécessite parfois l'utilisation de ressources propriétaires et non libres ([3]).

Concernant l'extraction des EN *entreprise*, Ahlers [2] a proposé une chaîne d'extraction très spécifique et liée aux Pages Jaunes. Cette chaîne s'appuie fortement sur des ressources propriétaires. Par ailleurs, la mine d'informations d'entreprises disponibles sur le web n'est que partiellement exploitées. En effet, la chaîne proposée s'appuie sur DMOZ qui ne contient que très peu de sites d'entreprises (moins d'1% des entreprises en Nouvelle-Aquitaine y sont référencées). En ce qui concerne l'extraction des événements, ce sont les approches basées sur l'apprentissage et les approches hybrides (apprentissage de patrons) qui sont privilégiées. Ceci se justifie sans doute par le fait qu'aucun standard n'organise le texte décrivant un événement, encore moins certaines de ses propriétés (le titre ou les noms d'artistes par exemple).

TABLE 5.1 – Synthèse des travaux relatifs à l'extraction de quelques types d'EN

Travaux	Approches mises en œuvre					EN ciblées				
	Patrons	Apprentissage	Ressources	Hybride	Temps	Lieu	Entité sociale	Entreprise	Événement	
Martins et al. [115]	✓		✓	✓	✓					
Filannino et al. [54]		✓			✓					
Alonso et al. [6]	✓				✓					
Le Parc-Lacayrelle et al. [98]	✓				✓					
Strotgen2010 et al. [171]	✓				✓					
Wang et al. [183]			✓			✓				
Berenike et al. [111]		✓				✓				
Taghva et al. [87]		✓				✓				
Nesi et al. [128]	✓					✓				
Li et al. [106]	✓					✓				
Alhers et al. [3]	✓					✓				
Schmidt et al. [162]	✓		✓	✓		✓				
Borges et al. [19]			✓			✓				
Yu et al. [188]	✓	✓	✓	✓		✓				
Alhers et al. [3]	✓					✓				
Shaan et al. [166]	✓		✓	✓			✓			
Le et al. [99]		✓					✓			
Ahlers et al. [2]	✓		✓	✓				✓		
Ritter et al. [149]		✓							✓	
Lin et al. [109]		✓							✓	
Li et al. [104]		✓							✓	
Lejeune et al. [101]	✓		✓	✓					✓	
Becker et al. [12]		✓							✓	
Kuzey et al. [94]	✓								✓	
Foley et al. [55]		✓							✓	
Serrano et al. [165]	✓	✓	✓	✓					✓	
Orlando et al. [134]	✓		✓	✓					✓	

Chapitre 6

Calcul de similarité entre entités nommées

Une même entité nommée (EN) peut être exprimée dans un texte par plusieurs syntagmes distincts. C'est le cas par exemple, de « *Mairie de Pau* » et « *Hôtel de ville de Pau* », qui font référence à la même entité physique. Ainsi, il est nécessaire de définir des fonctions permettant d'établir une telle similarité. Dans la pratique, ce calcul de similarité est utile pour deux opérations : (i) l'intégration sans doublon d'EN extraites dans une base de données ; (ii) l'appariement requête et EN en phase de recherche d'information (RI) [113]. De nombreux travaux se sont intéressés à ce problème. Dans ce chapitre, nous dressons une synthèse des propositions issues de ces travaux.

Nous organisons ces propositions en différentes approches que nous présentons dans la première section du chapitre. Ensuite, nous nous focalisons sur les applications de ces approches pour l'évaluation de la similarité entre EN.

6.1 Approches de calcul de similarité entre EN

Les approches de calcul de similarité entre EN diffèrent selon que l'on traite d'EN élémentaires ou d'EN complexes. Pour les EN élémentaires, ces approches dépendent des représentations choisies pour celles-ci : représentation textuelle, représentation sémantique et représentation numérique. Pour les EN complexes, le calcul de similarité se fait généralement en deux étapes : (i) la première calcule les similarités intermédiaires entre propriétés de même rang ; (ii) et la seconde agrège les scores intermédiaires obtenus pour en déduire la similarité globale. Considérons, par exemple, les deux POI ci-dessous constitués chacun de trois propriétés : le nom, la catégorie et les coordonnées géographiques respectivement.

$$p_1 = \langle \textit{Mairie de Pau}, \textit{Bâtiment institutionnel et administratif}, (43,29; -0,34) \rangle$$

$$p_2 = \langle \textit{Hôtel de ville de Pau}, \textit{Bâtiment institutionnel et administratif}, (43,29; -0,34) \rangle$$

Les catégories de ces POI sont représentées en exploitant celles de wikipedia⁷⁰. La première étape du calcul de similarité compare les noms des POI (« *Mairie de Pau* » et « *Hôtel de ville de Pau* »), leurs catégories (elles sont similaires pour l'exemple), et enfin les couples de coordonnées géographiques (similaires aussi). Ensuite la deuxième étape agrège les trois scores obtenus pour établir la similarité de ces deux POI.

70. https://fr.wikipedia.org/wiki/Catégorie:Bâtiment_institutionnel_et_administratif

Notons que certaines propriétés d'EN complexes peut être multivaluées, dans ce cas, elles sont représentées par des ensembles. Dans cette section, nous nous intéressons aussi au calcul de similarité entre ensembles.

6.1.1 Calcul de similarité basé sur la représentation textuelle des EN

Dans ce cas, le calcul de similarité se ramène à un problème de comparaison de textes (généralement courts : le texte d'une EN est constitué de quelques mots). La thèse de Nguyen [130] organise les approches permettant de faire cette comparaison en trois catégories principales : les approches basées sur les caractères, les approches basées sur les mots et les approches hybrides qui combinent les deux précédentes.

6.1.1.1 Approches basées sur les caractères

Ici, le texte est considéré comme un ensemble séquentiel de caractères. Le principe des approches de cette catégorie est de comparer les textes, en tenant compte de l'ordre des caractères. En général, les métriques proposées ici sont basées sur le nombre d'opérations élémentaires (insertion, suppression, transposition de caractères) nécessaires pour obtenir l'un des textes à partir de l'autre. Les métriques de Jaro [85], Jaro-Winkler [184] et Levenstein [103] sont les plus connues de cette catégorie. Leur efficacité est optimale sur les textes de longueur réduite [130].

6.1.1.2 Approches basées sur les mots

Contrairement aux approches basées sur les caractères, le texte ici est considéré comme un ensemble de mots. L'enjeu est de comptabiliser le nombre de mots identiques présents dans les deux textes sans toutefois tenir compte de leur ordre. Plusieurs métriques sont proposées pour cette catégorie d'approches, nous avons notamment la métrique de Jaccard et le cosinus TFIDF. Ces deux métriques sont adaptées aussi bien pour la comparaison de textes de longueur réduite, que pour les textes longs.

- **La métrique de Jaccard** [83] : elle est obtenue en évaluant le rapport de cardinalité de l'intersection des ensembles de mots et la cardinalité de leur union. Considérons deux textes S_1 et S_2 représentés par leurs ensembles de mots respectifs. La formule de Jaccard [130] qui évalue leur similarité est donnée par :

$$Jaccard(S_1, S_2) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|}$$

- **Le cosinus (TFIDF)** [155] : cette métrique est inspiré du modèle vectoriel de Salton utilisé en RI. Chaque texte est représenté par un vecteur de mots issus de l'union des ensembles correspondants aux textes comparés. Le poids d'un mot dans le vecteur représentant un texte dépend à la fois de sa fréquence dans celui-ci et de sa fréquence dans l'ensemble constitué des deux textes. Le score de similarité est obtenu en calculant le cosinus de l'angle formé par les deux vecteurs. Littéralement, si on note S_1 et S_2 les vecteurs représentant deux textes, leur similarité est calculée avec la formule :

$$TFIDF(S_1, S_2) = \cos(S_1, S_2) = \frac{S_1 \cdot S_2}{\|S_1\| \cdot \|S_2\|}$$

6.1.1.3 Approches hybrides

Dans les approches basées sur les mots, la comparaison entre mots est exacte et stricte. Considérons les textes : « ensembles de documents » et « ensemble de document ». Seul le mot « de » est dans leur intersection alors que leur réunion contient 5 mots distincts. Le score de similarité obtenu en utilisant la métrique de Jaccard pour ces deux textes est de 0,2 (1/5). Les deux vecteurs représentant ces textes sont (1, 1, 1, 0, 0) et (0, 1, 0, 1, 1) respectivement. Leurs normes sont égales et vaut $\sqrt{3}$, alors que le produit scalaire vaut 1. Au final le score de similarité entre ces deux textes, évalué avec le cosinus (TFIDF) vaut 0,33 (1/3). Les scores obtenus avec ces deux métriques sont très faibles au regard de la proximité entre les deux textes. En effet, tenir compte de la similarité entre caractères permettrait d'améliorer ce score : c'est la raison pour laquelle les approches hybrides ont été développées.

Dans ces approches hybrides, une première phase consiste à représenter les mots des deux textes selon des méta-caractères. Ensuite une seconde phase, applique les approches de comparaison basées sur les caractères pour ces méta-caractères. Parmi les métriques hybrides proposées, nous avons la métrique de Monge-Elkan [123], la métrique softTFIDF [32] et la métrique LIUPPA [130]. Les métriques de Monge-Elkan et softTFIDF ne tiennent pas compte de l'ordre des mots, tandis que la métrique LIUPPA, quant à elle, en tient compte. Ces trois métriques donnent toutes un score de similarité supérieur à 0,95 pour l'exemple précédent.

6.1.2 Calcul de similarité basé sur la représentation sémantique des EN

Les EN sont représentées dans ce cas en utilisant les concepts d'une ressource hiérarchique (ontologies, thesaurus). Le calcul du score de similarité entre deux concepts se fait en mesurant leur proximité sémantique. Trois stratégies sont utilisées [48] : (i) la première évalue la similarité en fonction du chemin le plus court entre les concepts ; (ii) la deuxième tient compte de la profondeur des concepts et/ou de leur plus proche ancêtre commun dans le graphe ; (iii) et la dernière exploite un corpus de documents et la fréquence d'occurrence des concepts dans ce corpus pour évaluer la similarité.

6.1.2.1 Calcul de similarité basé sur le chemin le plus court

Dans ce cas, évaluer la similarité entre deux concepts consiste tout d'abord à les projeter sur le graphe de la ressource, puis à mesurer le chemin le plus court entre les deux nœuds les représentant [112]. La mesure de similarité est fonction de la longueur de ce chemin. Il existe plusieurs métriques exploitant cette stratégie de calcul : la mesure de Rada [140], la mesure de Bulskov [23], etc.

6.1.2.2 Calcul de similarité basé sur la profondeur

Dans ce cas, une projection des concepts sur le graphe de la ressource est également requise. Le graphe est habituellement représenté par l'arbre qui représente la taxonomie (hiérarchie) de la ressource. Le score de similarité entre deux concepts est évalué en fonction des profondeurs respectives des nœuds les représentant par rapport au nœud racine de l'arbre. La taxonomie de la ressource est entièrement prise en compte ici, contrairement à la technique basée sur le chemin où juste la connectivité entre nœuds intervient. La mesure de Wu et Palmer [186] exploite cette stratégie, c'est l'une des plus utilisée dans la littérature. Soient deux concepts C_1 et C_2 de la ressource. On appelle C leur plus proche ancêtre commun dans la taxonomie de celle-ci. On note *profondeur* la fonction qui évalue la profondeur d'un

nœud (concept) dans l'arbre (nombre d'arêtes entre le nœud racine et ce dernier). La mesure de Wu et Palmer est donnée par la formule :

$$Wu_Palmer(C_1, C_2) = \frac{2 \cdot \text{profondeur}(C)}{\text{profondeur}(C_1) + \text{profondeur}(C_2)}$$

6.1.2.3 Calcul de similarité basé sur la fréquence d'occurrence dans un corpus

Le calcul du score de similarité entre concepts dans ce cas est fonction de la quantité d'information (contenu informationnel) relative à ces concepts dans un corpus donné. La notion de quantité d'information a été définie en théorie de l'information par Shannon et al. [167]. Dans le cadre du calcul de similarité sémantique, la quantité d'information d'un concept se calcule à partir d'une ressource et d'un corpus donné. Elle permet d'évaluer la pertinence du concept dans le corpus et elle est calculée en fonction de la fréquence d'occurrence de ses instances dans ce corpus. Considérons qu'on ait annoté n instances de concepts d'une ressource sémantique R dans un corpus, et que parmi les n annotations, n_C correspondent au concept C . La quantité d'information du concept C dans ce corpus est donné par la formule :

$$Q_R(C) = -\log\left(\frac{n_C}{n}\right)$$

La mesure de Resnik [147] évalue la similarité entre deux concepts comme la quantité d'information de leur plus proche ancêtre commun dans la taxonomie de l'ontologie. Ainsi deux couples de concepts (C_1, C_2) et (C_3, C_4) peuvent avoir le même ancêtre commun C , alors que C_1 est ancêtre de C_3 et C_2 celui de C_4 . Avec la mesure de Resnik, ces deux couples auraient le même score de similarité alors que les concepts ne sont pas au même niveau de la hiérarchie. C'est pour pallier cette limite, que des mesures comme celle de Lin [108] ont été définies. La mesure de Lin évalue la similarité entre deux concepts en fonction de leur quantité d'information individuelle et de celle de leur plus proche ancêtre commun.

6.1.3 Calcul de similarité basé sur la représentation numérique des EN

Dans ce cas, les EN sont représentées en utilisant des valeurs numériques. Dans la pratique, trois principaux cas sont à distinguer : les EN sont représentées sur un axe, les EN sont représentées dans le plan ou les EN sont représentées dans l'espace.

6.1.3.1 Représentation sur un axe

Ici, les EN peuvent correspondre à des points d'un axe, ou à des intervalles. Pour le cas où les EN sont représentées par des points, leur score de similarité est fonction de la distance qui les séparent [12]. Dans le cas où elles sont représentées par des intervalles (voir figure 6.1), leur similarité est calculée en fonction de la longueur de l'intervalle d'intersection (i) entre leurs représentations [98].

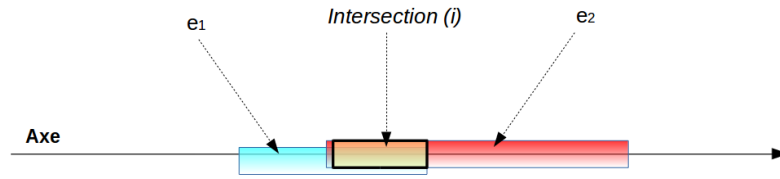


FIGURE 6.1 – Calcul de la similarité entre EN représentées sur un axe

6.1.3.2 Représentation sur un plan

Ici, les EN sont projetées sur un plan et elles sont représentées soit par des points, soit par des formes géométriques (voir figure 6.2). Ainsi, selon les cas possibles, différentes mesures de similarité sont proposées.

- Les deux EN correspondent à des *points* (figure 6.2.a) : dans ce cas, le calcul de similarité s'appuie sur des mesures de distance entre points du plan. La mesure la plus connue est la distance Euclidienne [42]. Pour les deux EN de la figure 6.2.a, elle est donnée par la formule :

$$distance(e_1, e_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

- Les deux EN correspondent à des *formes géométriques* (figure 6.2.b) : dans ce cas, le calcul de leur similarité est évalué en fonction de leur surface de recouvrement [91]. Des mesures de similarité comme celle de Walker et al. [180], Beard et Sharma [11] ou Sallaberry et al. [154] s'appuient sur cette technique. De même, la similarité entre formes géométriques peut aussi être calculée en fonction de la plus petite distance entre les points des contours de ces formes pris deux à deux. La mesure de Hausdorff [52] est basée sur cette technique et elle est adaptée au cas où les formes ne se touchent pas.
- l'une des EN est un *point* et l'autre une *forme géométrique* : dans ce cas, la non similarité est généralement établie si le point n'est pas inclus dans le polygone [152].

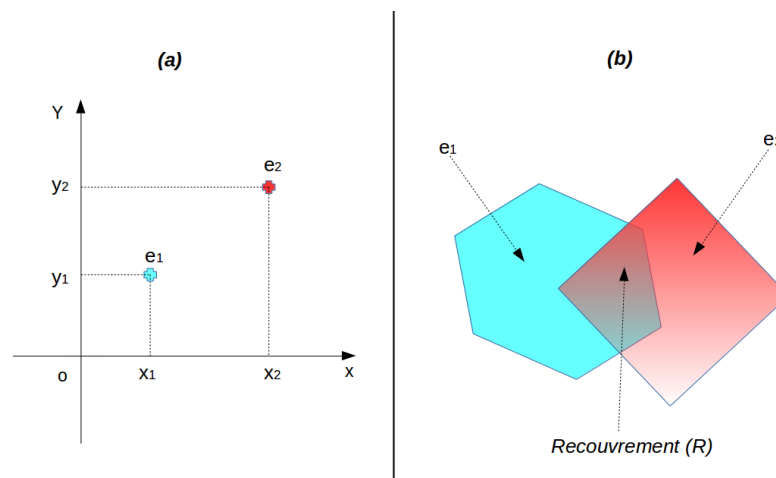


FIGURE 6.2 – Calcul de la similarité entre EN représentées sur un plan

6.1.3.3 Représentation dans l'espace

Pour ce dernier cas, les EN sont projetées dans l'espace. Ces EN correspondent habituellement à des points et l'enjeu est d'évaluer la distance entre ces points. La distance de Harversine [31], par exemple, permet d'évaluer la distance curviligne entre deux points P_1 et P_2 d'une sphère (S) (voir figure 6.3). On note (λ_1, ϕ_1) et (λ_2, ϕ_2) les couples de longitude et latitude en radians de ces points, et R le rayon de la sphère. La distance de Haversine entre P_1 et P_2 est donnée par la formule :

$$H(P_1, P_2) = 2.R.\arcsin\left(\sqrt{\sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\phi_1).\cos(\phi_2).\sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right)$$

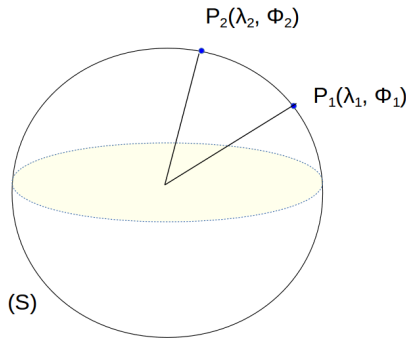


FIGURE 6.3 – Calcul de la similarité entre EN représentées par des points de l'espace

6.1.4 Calcul de similarité entre ensembles d'EN

Dans ce cas, les EN traitées sont multivaluées, chacune peut donc être représentée comme un ensemble d'EN, et l'enjeu ici est d'évaluer le degré de similarité entre ces deux ensembles. Une première approche consiste à utiliser la mesure de Jaccard [83], pour évaluer cette similarité. La formule de calcul est la même que celle définie en 6.1.1.2, à ceci près que l'on traite des ensembles d'EN en lieu et place de mots. Remarquons que quand la différence de cardinalité entre les deux ensembles est grande, même si l'un est inclus dans l'autre, la mesure de Jaccard donne un score faible.

Halkidi et al. [75] ont proposé une mesure de calcul de similarité entre ensembles de concepts d'ontologies. Cette mesure exploite les similarités des couples de concepts construits à partir du produit cartésien de ces ensembles. Leur objectif est de pallier les limites de la mesure de Jaccard lorsque la différence de cardinalités entre les ensembles à comparer est grande. Soient E_1 et E_2 deux ensembles de concepts contenant n et m éléments respectivement.

$$E_1 = \{u_1, u_2, \dots, u_n\}$$

$$E_2 = \{v_1, v_2, \dots, v_m\}$$

Le score de similarité entre E_1 et E_2 évalué par la mesure d'Halkidi s'appuie sur la mesure de Wu et

Palmer et elle est donné par la formule :

$$Halkidi(E_1, E_2) = \frac{1}{2} \left(\frac{1}{n} \sum_{i=1}^n \max_{1 \leq j \leq m} Wu_Palmer(u_i, v_j) + \frac{1}{m} \sum_{j=1}^m \max_{1 \leq i \leq n} Wu_Palmer(v_j, u_i) \right)$$

6.1.5 Calcul de similarité entre EN complexes

Le calcul de similarité entre EN complexes se fait généralement en deux phases comme décrit à la section 6.1. La première évalue les similarités entre propriétés de même rang (similarités intermédiaires) et la deuxième agrège les scores obtenus. La première phase exploite les approches présentées dans les sous-sections précédentes, pour évaluer les similarités intermédiaires. La difficulté se situe au niveau de la deuxième phase où il est question d'agrèger ces scores intermédiaires. Plusieurs approches ont été proposées pour réaliser cette agrégation. La plupart vient du domaine de la RI multicritères [135] et elles sont détaillées dans l'état de l'art de Marrara et al. [114].

Fox et Shaw [62] ont proposé quelques combineurs simples pour l'agrégation de scores intermédiaires dans le domaine de la RI multicritères : CombMAX, CombMIN, CombSUM, CombANZ et CombMNZ. CombMAX retourne le meilleur score obtenu pour les différents critères comme score global, contrairement à CombMIN qui retourne le plus petit score. CombSUM quant à lui retourne la somme des scores pour les différents critères. CombANZ divise la somme des scores intermédiaires par le nombre de scores non nuls alors que CombMNZ multiplie cette somme par le nombre de scores non nuls. Les valeurs obtenues avec ces combineurs peuvent être supérieures à 1. Lee et al. [100] recommandent donc de les normaliser pour faciliter les analyses.

Indépendamment de la RI multicritère, une des approches d'agrégation très utilisée dans la littérature est la combinaison linéaire. Cette approche consiste à calculer le score de similarité au niveau de l'EN complexe, en faisant la somme pondérée des différents scores de similarité entre propriétés (similarités intermédiaires). Les combineurs de Fox et Saw sont des cas particuliers de la combinaison linéaire. Plusieurs techniques sont utilisées pour déterminer les poids de similarité entre propriétés. Nous avons notamment la technique empirique, la technique basée sur des ordres de priorité et la technique basée sur les méta-heuristiques.

Pour la technique empirique, l'expert donne des poids aux similarités intermédiaires selon qu'il veut donner plus d'importance à certaines d'entre elles. Cette technique est utilisée, par exemple, par Wongsuphasawat et al. [185] pour le calcul de similarité entre séquences d'événements.

Concernant la technique basée sur les ordres de priorité, Da Costa Pereira et al. [35] ont proposé une approche d'agrégation qu'ils ont appelé *Prioritized Aggregation*. Pour cette approche, les différentes propriétés de l'EN sont ordonnées selon des niveaux d'importance décroissant. Le poids de chaque similarité dans le combinaison linéaire est déterminé en fonction de sa valeur pour les EN comparées et du niveau d'importance de la propriété associée. Considérons, par exemple, qu'on s'intéresse au calcul de similarité entre EN complexes ayant chacune trois propriétés. On note s_1 , s_2 et s_3 les similarités des propriétés au rang 1, 2 et 3 respectivement et w_1 , w_2 et w_3 leurs poids dans la combinaison linéaire. On suppose que la similarité de la propriété 3 est la plus importante, suivie de celle de la propriété 1. D'après l'approche *Prioritized Aggregation*, $w_1 = 1$, $w_2 = w_1.s_1$ et $w_3 = w_2.s_2$. Cette technique d'agrégation permet de pénaliser les critères les moins importants, lorsque ceux les plus importants ne sont pas satisfaits. Remarquons, cependant que les poids des similarités intermédiaires varient en fonction des EN comparées.

Enfin, pour la technique basée sur les méta-heuristiques, des algorithmes d'optimisation sont utilisés

pour déterminer les poids des similarités intermédiaires dans la combinaison linéaire. Khrouf et al. [89], par exemple, utilisent la technique d’optimisation par essais particuliers [46] pour le calcul des poids des similarités intermédiaires. Nikolov et al. [131], quant à eux, exploitent les algorithmes génétiques pour la détermination des ces poids dans le domaine du « data linkage » [53]. Il faut noter que les méta-heuristiques n’ont de l’intérêt que lorsque les EN ont un nombre important de propriétés.

Une autre approche d’agrégation des scores intermédiaires est basée sur les fonctions logiques. En effet, une expression logique est construite pour chaque propriété et des opérateurs logiques (ET, OU, NON, etc.) sont utilisés pour les fusionner [165]. La similarité obtenue est booléenne : soit les EN sont similaires, soit elle ne le sont pas. Considérons deux EN complexes e_{c_1} et e_{c_2} constituées de 3 propriétés. On note s_1, s_2, s_3 les scores intermédiaires pour ces trois propriétés et ϵ_1, ϵ_2 et ϵ_3 les seuils de similarité associés. Un exemple de fonction d’agrégation serait :

$$\text{similarite}(e_{c_1}, e_{c_2}) = (s_2 > \epsilon_2) \quad \text{OU} \quad ((s_1 > \epsilon_1) \quad \text{ET} \quad (s_3 > \epsilon_3))$$

Pour cet exemple, la similarité est établie si le score intermédiaire correspondant à la propriété au rang 2 est supérieur au seuil ϵ_2 ou si les scores correspondant aux propriétés 1 et 3 sont supérieurs à leurs seuils respectifs.

6.2 Travaux dédiées au calcul de similarité entre EN

Les différentes approches présentées dans la section précédente sont mises en œuvre dans plusieurs travaux pour évaluer la similarité entre EN. Dans cette section, nous en présentons quelques uns que nous organisons selon les types d’EN traités : les EN temporelles, les EN spatiales, les entités sociales (personnes ou organisations), les EN de type POI et les EN événement.

6.2.1 Calcul de similarité entre EN temporelles

Le Parc-Lacayrelle et al. [98] proposent dans un contexte de RI, une fonction pour évaluer le score de similarité entre deux EN temporelles. Cette proposition se base sur la représentation numérique, sous forme d’intervalles, des EN temporelles sur l’axe du temps. Le calcul du score de similarité se fait par évaluation de la longueur de l’intervalle d’intersection proportionnellement aux intervalles représentant les 2 EN temporelles observées.

Becker et al. [12] proposent également une métrique pour l’évaluation de la similarité en EN temporelles, représentées sur l’axe du temps par des points. Cette métrique tient compte d’un seuil au delà duquel deux EN sont considérées non similaires, nous le noterons θ . Soient deux EN temporelles e_1 et e_2 , et on note $|e_2 - e_1|$ la longueur du segment les reliant. La mesure de similarité entre ces deux EN est donnée par :

$$\text{Becker}(e_1, e_2) = \begin{cases} 0 & \text{si } |e_2 - e_1| > \theta \\ 1 - \frac{|e_2 - e_1|}{\theta} & \text{sinon.} \end{cases}$$

Serrano et al. [165] proposent aussi une approche pour le calcul de similarité entre dates. Chaque date est représentée comme une EN complexe composée de trois propriétés : le jour, le mois et l’année. La fonction de calcul de similarité se contente de comparer les différentes propriétés entre elles et les agrègent en utilisant des fonctions logiques. Pour avoir identité entre EN, toutes les propriétés renseignées doivent

être identiques.

6.2.2 Calcul de similarité entre EN spatiales

Dans le cas où ces EN sont représentées comme des concepts sémantiques, Hastings [80] propose une fonction qui exploite la profondeur des concepts dans le graphe par rapport à leur plus proche ancêtre commun. Si on note d_1 et d_2 ces profondeurs respectives pour les concepts associés aux EN e_1 et e_2 , la similarité est calculée avec la fonction :

$$\text{similarite}(e_1, e_2) = 2^{-(d_1+d_2)}$$

Janowicz et al. [84] proposent également une approche de calcul de similarité entre EN spatiales en exploitant leur représentation sémantique. Un premier score de similarité évalue la proximité sémantique des concepts en utilisant la mesure de Wu et Palmer. Un deuxième score évalue la similarité entre attributs des concepts dans la ressource. Les deux scores obtenus sont agrégés en faisant une combinaison linéaire dont les poids sont déterminés de façon empirique.

Plusieurs travaux de calcul de similarité entre EN spatiales exploitent la représentation numérique de celles-ci. Pour le cas où les EN sont représentées :

- par des polygones sur un plan, Wang et al. [182] calculent leur surface de recouvrement pour évaluer la similarité. Des techniques plus sophistiquées comme celle de Sallaberry et al. [154] exploitent également la surface de recouvrement, mais en tenant compte des proportions de chacun des polygones ;
- par des points (latitude et longitude) du plan, Mckenzie et al. [117] utilisent la distance euclidienne pour le calcul de leur similarité ;
- sur un plan, l'un par un point et l'autre par un polygone, la non similarité est habituellement établie dès que le point n'est pas inclus dans le polygone. Rueben [152] utilisent une fonction binaire qui donne un score de 1 si le point est dans le polygone et 0 sinon ;
- par des points sur la surface de la terre, Bahram et al. [9] exploitent la distance de Harversine pour évaluer leur similarité.

6.2.3 Calcul de similarité entre EN sociales

Les EN *sociales* sont généralement exprimées par des syntagmes de quelques mots (nom de personnes, d'organisations, etc.). Des ressources sémantiques sont parfois mobilisées pour les représenter. De ce fait, le calcul de leur similarité repose soit sur les approches de comparaison de textes, soit sur des approches de calcul de similarité basées sur une représentation sémantique. Khrouf et al. [89], par exemple, utilisent les ressources du web sémantique (DBpedia, Music Brainz⁷¹, etc.) pour évaluer la similarité entre agents intervenant dans un événement. Les noms d'agents sont projetés sur ces ressources pour identifier les concepts correspondants. Chaque concept est une EN décrite par un ensemble de propriétés (le nom de l'agent, un résumé court, etc.) extraites de la ressource. Le calcul de la similarité se fait en deux étapes. La première évalue la similarité entre propriétés de même rang en comparant les textes, et la seconde agrège les résultats obtenus en utilisant une approche basée sur des fonctions logiques.

71. <https://musicbrainz.org/>

Dans le même ordre d'idées, Serrano et al. [165] utilisent une approche basée sur la comparaison de textes pour évaluer la similarité entre noms de personnes. Le calcul de la similarité se fait en deux étapes également. Une première se contente d'évaluer la similarité entre les représentations textuelles des deux EN. C'est la distance de Jaro qui est utilisée pour cette comparaison. Si la distance résultante est supérieure à un seuil, la similarité est directement établie. Sinon une deuxième étape compare la représentation textuelle de l'une, avec un ensemble de noms alternatifs pour l'autre, obtenus en exploitant une ressource du web sémantique (DBpedia). Si au moins l'un des noms alternatifs de la deuxième EN est similaire au texte de la première, alors la similarité est également établie. Dans tous les autres cas, il n'y a pas de similarité.

Moreau et al. [124], quant à eux, ont expérimenté plusieurs métriques de comparaison de textes pour évaluer la similarité entre noms de personnes et d'organisations. Ils évaluent notamment les approches basées sur les caractères (Jaro), celles basées sur les mots (TFIDF) et les approches hybrides (softTFIDF). Ils ont démontré que la métrique softTFIDF donne les meilleurs résultats.

Enfin les travaux de Shet et al. [168] exploitent la représentation sémantique des acteurs du système éducatif pour évaluer leur degré de similarité. Cette proposition s'appuie sur une ontologie décrivant les différents acteurs et leurs relations (professeur, étudiant, diplômé, etc.). Le calcul de la similarité ré-utilise la métrique de Wu et Palmer.

6.2.4 Calcul de similarité entre EN de type POI

Scheffer et al [161] ont pour objectif de consolider les données de POI provenant de plusieurs réseaux sociaux. Ainsi, étant donné un POI, leur problème est d'identifier tous ceux qui lui sont similaires pour fusionner leurs informations (*data linkage*). Pour ce faire, une première étape exploite la représentation numérique sous forme de point de la propriété *spatiale* du POI, pour identifier tous ceux qui sont situés dans un rectangle précis autour de celui-ci. C'est la distance euclidienne qui est utilisée pour cette phase. La deuxième étape exploite la représentation textuelle de la propriété *nom* du POI, pour identifier ceux qui lui sont similaires à partir de la short-liste constituée. C'est la distance de Levenstein qui est utilisée pour cette deuxième phase. Il faut noter que dans leur contexte de travail, les deux propriétés (*spatiale* et *nom*) sont toujours renseignées.

Kim et al. [91] s'intéressent également au calcul de similarité entre POI décrit dans le texte. Le calcul de la similarité globale s'appuie sur trois similarités intermédiaires. La première similarité correspond à celle des noms de POI, elle est évaluée en utilisant la distance de Levenstein. La deuxième similarité porte aussi sur les noms de POI, mais cette fois en utilisant WordNet pour remplacer certains mots par des synonymes. La dernière similarité exploite les relations (à côté de, inclus dans, etc.) entre lieux pour évaluer leur similarité. Les relations entre lieux sont décrits dans une ressource. Les trois similarités sont agrégées en utilisant la combinaison linéaire, avec des poids déterminés de façon empirique.

6.2.5 Calcul de similarité entre EN événement

Plusieurs travaux se sont intéressés au calcul de similarité entre événements. Railean et al. [142], par exemple, proposent une technique d'évaluation de la similarité entre un événement social et un tweet. Dans leur proposition, la similarité globale est évaluée à partir de plusieurs similarités intermédiaires : la similarité entre le tweet et les acteurs de l'événement ; la similarité entre le tweet et le titre de l'événement ; la similarité entre le tweet et le nom du lieu de l'événement ; enfin la similarité entre le tweet et la

description de l'événement. La similarité globale est obtenue par combinaison linéaire des différentes similarités intermédiaires, avec des poids déterminés de façon empirique. Notons que toutes les similarités intermédiaires sont calculées en utilisant des approches basées sur la comparaison de textes.

Serrano et al. [165] proposent également d'agréger les similarités entre propriétés de même rang en utilisant une combinaison linéaire pour évaluer la similarité entre événements. Les poids des similarités intermédiaires sont déterminés de façon empirique. Dans cette proposition, la similarité temporelle est calculée en utilisant l'approche détaillée en 6.2.1. La similarité spatiale est basée sur la représentation sémantique des EN : c'est la ressource GeoNames⁷² qui est exploitée à cet effet. L'approche de calcul de similarité mise en œuvre exploite le chemin entre concepts. Le calcul de similarité entre acteurs d'événements utilise l'approche détaillée en 6.2.3.

Khrouf et al. [89] s'intéressent à la réconciliation d'événements provenant de plusieurs sources du web. Ces événements sont extraits en interrogeant les API des différentes sources puis sont représentées suivant l'ontologie LODE [176]. Leur contribution vise l'interconnexion d'événements similaires dans l'ontologie. De ce fait, ils s'appuient sur une opération de calcul de similarité. Dans leur approche, ce calcul exploite la combinaison linéaire des similarités de propriétés d'événements. Contrairement aux deux exemples précédents, les poids ici ne sont pas déterminés de façon empirique, mais en se basant sur une méta-heuristique. C'est la technique d'optimisation par essaims (PSO) [46] qui permet de déterminer les poids de la combinaison linéaire.

Dans un contexte de détection d'événements, Becker et al. [12] proposent une approche de calcul de proximité entre documents, basée sur les descripteurs d'événements qu'ils contiennent. En effet, leur approche consiste à regrouper le flux de documents provenant des réseaux sociaux, en groupes susceptibles de décrire un même événement. Chaque document est caractérisé par un ensemble de propriétés : le titre, les tags, la description, le lieu, la date, etc. La similarité des documents pour une propriété donnée est évaluée en utilisant du clustering. En fait, l'ensemble des documents est segmenté pour chaque propriété en clusters. Deux documents sont similaires pour une propriété, s'ils sont dans le même cluster pour ladite propriété et un score de 1 est associée à la similarité intermédiaire. La similarité globale au niveau des documents est obtenue aussi en faisant la combinaison linéaire des similarités intermédiaires, avec des poids déterminés de façon empirique.

6.3 Bilan

Dans ce chapitre, nous avons présenté les principales approches utilisées pour calculer des scores de similarité entre EN. Le calcul en lui-même dépend du mode de représentation choisi.

En ce qui concerne le calcul de similarité entre EN élémentaires, plusieurs approches existent, et le tableau 6.1 en dresse la synthèse. Lorsque les EN sont représentées :

- sous forme textuelle, les métriques hybrides sont les plus adaptées parce qu'elles comparent à la fois les mots et les caractères ([130], [124]) ;
- par des concepts sémantiques, l'approche mise en œuvre pour le calcul de similarité dépend du contexte. Si la comparaison des concepts est fonction du corpus dans lequel ils sont extraits, l'approche basée sur la quantité d'information nous semble la plus adaptée. Dans le cas où on souhaite exploiter uniquement la taxonomie de la ressource pour le calcul de similarité, la mesure

72. <http://www.geonames.org/>

de Wu et Palmer est généralement privilégiée parce qu'elle tient compte à la fois du plus proche ancêtre commun des concepts et de leurs profondeurs ;

- sous forme numérique par des intervalles sur un axe (EN temporelles), le calcul de l'intervalle d'intersection permet d'évaluer la longueur du segment commun entre les EN. D'où sa pertinence dans ce cas ([98], [173]). Lorsque les EN sont plutôt représentées par des points sur un axe, l'évaluation de la longueur du segment entre eux est adapté pour mesurer leur similarité ([12]) ;
- sous forme numérique par des points du plan ou de l'espace, les mesures basées sur le calcul de distance permettent d'évaluer leur similarité ([117], [9]). Dans le cas, où les EN sont représentées par des polygones, le calcul de la surface de recouvrement ([182]) est suffisant si la proportionnalité des surfaces est respectée. Cependant, le recouvrement seul ne suffit plus dans le cas contraire, d'où la nécessité d'utiliser des mesures qui tiennent compte de la proportion des surfaces de polygones ([81], [154]). Dans le cas enfin où les EN sont représentées l'une par un point et l'autre par un polygone, l'approche proposée par Ruben et al. [152] paraît assez naïve. Elle se contente d'établir la similarité si le point est dans le polygone, sans toutefois tenir compte du cas où le point pourrait se trouver dans le voisinage de celui-ci ;
- sous forme d'ensembles, la mesure de Jaccard [83] est adaptée lorsque les deux ensembles ont des cardinalités proches. Dans le cas contraire, la mesure d'Halkidi et al. [75] évalue mieux la similarité. Ces mesures peuvent être utilisées pour évaluer la similarité entre ensembles d'EN élémentaires ou entre ensembles d'EN complexes.

En ce qui concerne le calcul de similarité entre EN complexes (constituées de plusieurs propriétés), nous avons vu que l'enjeu était d'agréger les scores de similarité intermédiaires (similarité entre propriétés de même rang). Le tableau 6.2 présente quelques travaux traitant de techniques d'agrégation. Nous constatons que la combinaison linéaire est l'approche la plus utilisée pour agréger les similarités intermédiaires. Ceci s'explique par le fait que ce soit une approche simple à mettre en œuvre et pour les travaux qui l'exploitent, ils supposent qu'il n'a pas de corrélation entre similarités de propriétés. Notons également que les poids des différentes similarités intermédiaires sont généralement déterminées de façon empirique. Concernant l'approche basée sur les fonctions logiques, elle est adaptée pour deux cas : (i) lorsqu'on veut comparer une EN à plusieurs autres, dans ce cas, les fonctions logiques permettent de procéder par élimination ([161]) ; (ii) dans le cas où la similarité ou la non similarité peut être établie juste en comparant certaines propriétés ([89]). Dans ces deux cas, il n'est pas nécessaire de calculer toutes les similarités intermédiaires comme cela est le cas pour la combinaison linéaire.

TABLE 6.1 – Synthèse des travaux relatifs au calcul de similarité entre EN élémentaires

Travaux	Approches selon la représentation										EN ciblées		
	Rep. textuelle			Rep. sémantique			Rep. numérique		Temps	Lieu	EN Sociale		
	Caractères	Mots	Hybride	Chemin	Profondeur	Axe	Plan	Espace					
Le Parc. et al. [98]						✓			✓				
Becker et al. [12]						✓			✓				
Wang et al. [182]							✓				✓		
Nguyen et al. [130]			✓								✓		
Teissière et al. [173]						✓			✓				
Hill et al. [81]								✓			✓		
Rueben et al. [152]								✓			✓		
Sallaberry et al. [153]								✓			✓		
Moreau et al. [124]	✓	✓	✓									✓	
Bahram et al. [9]											✓		
Shet et al. [168]					✓							✓	
Mckenzie et al. [117]								✓				✓	

TABLE 6.2 – Synthèse des travaux relatifs au calcul de similarité entre EN complexes

Travaux	Agrégation des scores intermédiaires			EN ciblées					
	Combinaison Linéaire		Logique	Temps	Lieu	EN Sociale	POI	Événement	
	Empirique	Meta-heuristique							
Railean et al. [142]	✓							✓	
Serrano et al. [165]	✓		✓	✓		✓		✓	
Janowicz et al. [84]	✓				✓				
Scheffer et al. [161]			✓				✓		
Khrouf et al. [89]						✓			
Kim et al. [91]	✓						✓		
Wongsuphasawat et al. [185]	✓							✓	
Becker et al. [12]	✓							✓	

Bilan de l'état de l'art

Dans le cadre du projet *Cognisearch*, notre objectif est de développer des services de recherche d'informations portant sur des EN *entreprise* et *événement*, en exploitant le contenu du web et des ressources non propriétaires. La réalisation de cet objectif, que nous voulons généraliser à tous les types d'EN complexes, soulève de nombreux problèmes qui s'articulent autour de trois axes notamment : (i) la représentation des EN complexes ; (ii) l'extraction d'EN complexes à partir du contenu textuel du web ; (ii) et enfin le calcul de similarité entre EN complexes ;

Représentation des EN

Concernant la représentation des EN plusieurs modèles ont été proposés. Le modèle *TIMEX3*, par exemple, semble adapté dans notre contexte pour la représentation des informations temporelles. Concernant certaines informations thématiques comme les catégories d'événements, les activités d'entreprises ou leurs métiers, une représentation sémantique de celles-ci serait convenable. Pour la représentation d'adresses, le modèle de l'Etalab, bien que focalisé uniquement sur l'aspect spatial, permet de représenter un grand nombre de propriétés d'adresses. Nous envisageons donc de le ré-utiliser et de le compléter pour représenter nos adresses. En ce qui concerne les entreprises, nous comptons ré-utiliser le modèle SI-RENE comme modèle de base et nous projetons de le compléter pour prendre en compte les nombreuses informations d'entreprises extraites du web (les différentes localisations, les métiers, les activités, les produits ou même les services). Enfin pour la représentation des événements, tous les modèles existants sont basés sur une représentation à plat. Ceci engendre des redondances lorsqu'on traite d'événements socio-culturels, pour lesquels, le même événement peut avoir plusieurs représentations programmées à différents lieux, dates et horaires. De ce fait, il serait intéressant d'avoir plutôt un modèle factorisé pour limiter les redondances.

Extraction d'EN complexes sur web

Concernant l'extraction d'information sur le web à partir du texte des pages, plusieurs difficultés sont à prendre en compte. Nous avons notamment, l'identification des pages pertinentes à traiter, le contexte bruité caractérisant certaines propriétés, l'expression de plusieurs propriétés par des formes non régulières, la sur-composition des syntagmes correspondant à des propriétés. Pour la première difficulté, nous envisageons ré-utiliser certains prétraitements présentés en section 5.1, notamment le filtrage du web et la classification des pages web.

Concernant la difficulté liée au contexte bruité, très peu de travaux s'y sont intéressés. Nous rappelons

que dans un tel contexte, le texte analysé peut contenir plusieurs informations candidates, susceptibles de correspondre à une propriété d'EN complexes recherchées. Pour notre contexte, la proposition d'Orlando et al. [134] détaillée en section 5.3.5.2 ne semble pas adaptée, au regard du grand nombre d'EN erronées qu'elle annoterait.

En ce qui concerne la difficulté liée au fait que certaines propriétés ne suivent pas formes régulières d'écriture, plusieurs travaux, majoritairement basés sur l'apprentissage ont été proposés pour le résoudre ([99], [55], etc.). De même, des ressources sont parfois mobilisées pour l'identification de celles-ci, lorsqu'elles sont décrites dans des ressources. Pour notre problème, nous envisageons également de ré-utiliser ces approches pour annoter les textes.

Concernant la difficulté liée à la sur-composition de syntagmes relatif à certaines EN, des patrons ou l'apprentissage sont utilisés pour identifier l'EN globale ([64], [106], [87]). Nous comptons également ré-utiliser ces approches pour notre problème.

Enfin, nous constatons que toutes les chaînes d'extraction d'EN complexes proposées dans l'état de l'art sont ad-hoc et propres aux types EN ciblés. Une des contraintes fixées par *Cogniteev* est la généralité des traitements, de ce fait, nous envisageons une chaîne générique, valide pour tout type d'EN complexes. Celle-ci devra permettre d'identifier correctement les propriétés de l'EN, en tenant compte des verrous énoncés précédemment.

Calcul de similarité entre EN complexes

Concernant le calcul de similarité entre EN complexes, nous avons vu qu'il se fait habituellement en deux étapes : le calcul de similarité entre propriétés de même rang et l'agrégation des scores obtenus. Pour cette agrégation, l'approche la plus utilisée est la combinaison linéaire. Les poids sont généralement déterminés de façon empirique, ce qui constitue une limite dans la mesure où ce choix est très subjectif et dépend fortement de l'expert et non des données. Certains travaux comme ceux de Khrouf et al. [89] ou de Nikolov [131], par exemple, exploitent des méta-heuristiques pour la détermination des poids. L'utilisation de ce type d'approches est efficace quand les EN ont un grand nombre de propriétés. Nous souhaitons mettre sur pied des approches qui restent efficace pour le calcul de similarité entre EN avec peu de propriétés comme pour celles avec un grand nombre de propriétés. Nous envisageons également des approches basées sur la combinaison linéaire, mais avec des techniques de détermination des poids automatiques ou semi-automatiques en fonction des données traitées et qui restent efficaces pour des EN avec peu de propriétés. Par ailleurs, l'approche par combinaison linéaire suppose une indépendance a priori des similarités entre propriétés, ce qui n'est pas toujours garanti. En effet, pour des EN *entreprise*, par exemple, la similarité entre produits fabriqués pourraient être corrélée à la similarité entre activités exercées. Afin de tenir compte d'éventuelles corrélations, nous songeons à des approches de calcul de similarité entre EN complexes, qui ne reposent sur aucune hypothèse d'indépendance.

Dans la partie suivante, nous présentons en détail nos différentes propositions pour les problèmes soulevés. Ces propositions sont expérimentées et mises en œuvre au travers de deux prototypes des services *Cognisearch Business* et *Cognisearch Event*.

Troisième partie

Contributions à l'extraction et au calcul de similarité entre entités nommées complexes

Chapitre 7

Architecture *Cognisearch*

La problématique traitée dans le cadre du projet *Cognisearch* vise principalement l'analyse du contenu du web pour extraire des informations relatives à des EN complexes. Ces informations sont ensuite consolidées et structurées dans des index. Enfin, ces index sont interrogés pour répondre à des besoins d'information. Pour résoudre cette problématique, nous avons conçu une architecture (*Cognisearch*), dédiée à la mise en œuvre des traitements décrits en figure 7.1. Cette figure présente l'architecture *Cognisearch* avec ses principaux modules de traitement, les données manipulées et les ressources mobilisées pour son fonctionnement. Cette architecture est générique et elle sera mise en œuvre pour le traitement des deux types d'EN auxquelles nous nous intéressons : les EN *entreprise* et les EN *événement*.

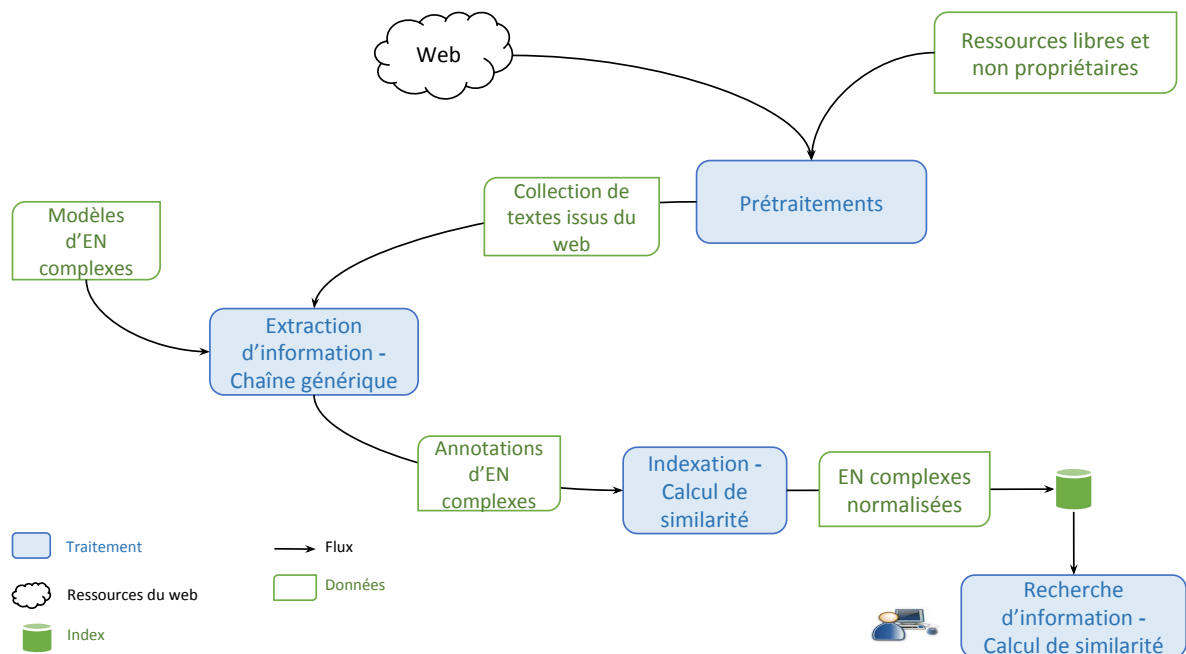


FIGURE 7.1 – Architecture *Cognisearch*

Le premier module, nommé « *Prétraitements* », exploite des ressources libres et non propriétaires pour

traiter le contenu du web et constituer une collection de textes. Son objectif de filtrer le sous-ensemble du web pertinent pour l'extraction des EN ciblées. Ensuite, seul le texte des pages de ce sous-ensemble est analysé pour l'extraction d'information.

Le deuxième module, nommé « *Extraction d'information* », est dédié à l'analyse des textes obtenus en sortie du module de « *Prétraitements* », pour annoter des EN complexes. Ce traitement intègre notre première contribution : **une chaîne générique d'extraction d'EN complexes dans un texte**. Cette chaîne prend en entrée, un texte et des ressources dont le modèle décrivant les propriétés de l'EN ciblée. Ce modèle permet de spécifier les informations à identifier dans le texte.

Les informations extraites sont transformées, normalisées et structurées pour alimenter les index dans le troisième module, nommé « *Indexation* ». Pour garantir la cohérence des index, ce module exploite notre deuxième contribution : **des approches génériques de calcul de similarité entre EN complexes**. Ces approches permettent d'intégrer les EN extraites dans les index en évitant les doublons ou en enrichissant les EN déjà indexées.

Le dernier module, nommé « *Recherche d'information* », prend en entrée un besoin d'information et analyse les index pour rechercher les EN répondant à ce besoin. Le processus d'analyse des index pour l'identification des EN pertinentes exploite également notre contribution relative au **calcul de similarité entre EN complexes**

Le chapitre 8 présente les modèles que nous avons définis pour représenter les EN *adresse*, *entreprise* et *événement*. Au chapitre 9, nous décrivons notre chaîne générique d'extraction d'EN complexes avant de l'instancier pour extraire des EN *adresse*, *entreprise* et *événement*. Le chapitre 10 détaille nos approches pour le calcul de similarité entre EN complexes. Le chapitre 11 enfin, décrit la mise en œuvre de l'architecture *Cognisearch* pour la conception et la réalisation de prototypes des deux services demandés par *Cogniteev*.

Chapitre 8

Modèles de représentation d'entités nommées complexes

L'architecture *Cognisearch* vise l'extraction et l'indexation d'EN complexes à partir du contenu du web pour alimenter des services de recherche d'information. À cet effet, il est primordial de définir quelles sont les informations qui constituent ces EN. Nous avons de ce fait défini des modèles qui serviront de référence de représentation d'EN complexes.

Nous nous intéressons plus précisément à trois types d'EN complexes : les EN *adresse*, *entreprise* et *événement*. Nous avons vu dans l'état de l'art que plusieurs modèles ont été proposés pour les représenter. Cependant, ceux-ci ne sont pas tout à fait adaptés pour notre problème pour deux principales raisons : (i) ils ne représentent pas toutes les propriétés ciblées par *Cognisearch* ; (ii) ils introduisent parfois des redondances.

Nous présentons dans ce chapitre nos modèles de représentation d'EN *entreprise* et *événement*. Pour notre projet, nous avons besoin de connaître les localisations d'entreprises et les lieux des événements. Nous représentons ces informations par des adresses, de ce fait, nous avons également défini un modèle d'adresse.

8.1 Modèle de représentation d'EN *adresse*

Dans l'état de l'art, plusieurs modèles ont été proposés pour représenter les adresses. Nous avons notamment le modèle Adresse de l'Etalab⁷³ et le modèle *h-adr*⁷⁴ de microformats.org. Considérons les deux exemples d'adresses ci-dessous.

Exemple 1 :

1	Résidence Isaac Newton
2	2 Bis, Rue de la sous-préfecture,
3	CS 38750,
4	34700, Lodève CEDEX 07, France.

73. <https://adresse.data.gouv.fr/pdf/description-contenu.pdf>

74. <http://microformats.org/wiki/h-adr>

5

Exemple 2 :

1 ZA Les Pins Verts
 2 33650 Saucats.
 3

Dans ces exemples, « *Résidence Isaac Newton* » et « *ZA Les Pins Verts* » correspondent à des compléments d'adresse. Le modèle de l'Etalab, ne permet pas de les représenter alors que la propriété *p-extended-address* du modèle *h-adr* peut être utilisée à cet effet. Par ailleurs, dans le modèle *h-adr*, toutes les informations de la voie pour l'exemple 1 (« *2 Bis, Rue de la sous-préfecture* »), sont représentées en une seule propriété (*p-street-address*), alors que le modèle *Adresse* de l'Etalab utilise trois propriétés distinctes : *numero_voie* (*2*), *rep* (*Bis*) et *nom_voie* (*Rue de la sous-préfecture*). Toutefois, aucun de ces deux modèles ne permet de représenter des informations comme le numéro de course spéciale⁷⁵ (*CS 38750*).

Notre modèle d'EN *adresse* (illustré figure 8.1) combine le modèle de l'Etalab avec celui de *h-adr*. Il intègre, par ailleurs, de nouvelles propriétés notamment le numéro de course spéciale, le nom du département ou même numéro de courrier. Cet ajout est justifié par le fait que nous avons constaté que ces propriétés sont parfois spécifiées pour les adresses contenues dans les pages web de notre corpus. La représentation des coordonnées géographiques de l'adresse s'appuie sur le type *GeoPoint*. Ce type permet de spécifier la longitude, la latitude et l'altitude d'un point.

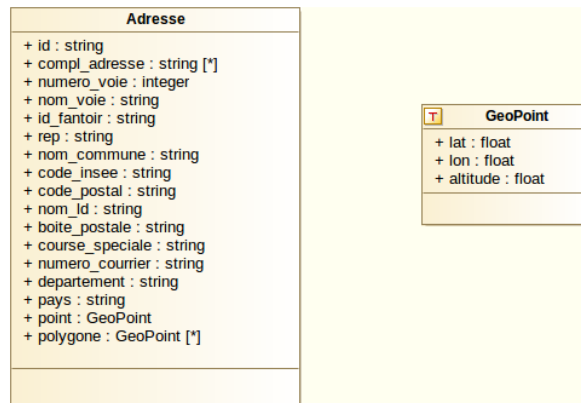


FIGURE 8.1 – Notre Modèle de représentation d'EN *adresse*

Certaines adresses, comme celle de l'exemple 2, n'ayant pas assez d'informations pour être représentées par un point, nous les représentons par un polygone correspondant à la commune de l'adresse. Ce polygone est défini par une liste de points correspondant à ses différents sommets.

Le tableau 8.1 définit chaque propriété de notre modèle, en spécifiant pour chacune sa source. Les propriétés que nous avons ajoutées ont pour source *Cognisearch*.

75. <https://www.service-public.fr/particuliers/actualites/A10432>

Propriétés	Définition	Source
id	Identifiant de l’adresse dans la base nationale d’adresses (BAN)	Etalab
compl_adresse	Compléments d’adresse facilitant sa localisation	h-adr
numero_voie	Numéro de la voie de circulation	Etalab
nom_voie	Nom de la voie de circulation	Etalab
id_fantoir	Identifiant de la voie dans le répertoire FANTOIR ⁷⁶	Etalab
rep	Indice de répétition associé au numéro de voie (Bis, Ter, ...)	Etalab
nom_commune	Nom officiel de la commune	Etalab
code_insee	Identifiant de la commune suivant le Code Officiel Géographique	Etalab
code_postal	Code postal associé à la commune	Etalab
nom_ld	Nom du lieu-dit s’il y en a un	Etalab
boite_postale	Numéro de boîte postale	h-adr
course_speciale	Numéro de course spéciale pour la poste	Cognisearch
numero_courrier	Numéro de courrier (introduit par « CEDEX » ou « CIDEX »)	Cognisearch
departement	Département dans lequel se situe la commune	Cognisearch
pays	Pays dont fait partie la commune	Cognisearch
point	Coordonnées géographiques de l’adresse (latitude, longitude et altitude) dans le système WGS 84 ⁷⁷	Cognisearch
polygone	Polygone représentant une commune	Cognisearch

TABLE 8.1 – Propriétés d’une adresse selon notre modèle

Propriétés	Valeurs
id	ADRNIVX 0000000322655865
numero_voie	2
nom_voie	Rue de la sous-préfecture
id_fantoir	1060
rep	Bis
nom_commune	Lodève
code_insee	34142
code_postal	34700
nom_ld	LES CHARMETTES
point	(3,31942391511594, 43,7335033547543, 172,62)
polygone	-
compl_adresse	Résidence Isaac Newton
boite_postale	-
course_speciale	CS 38750
numero_courrier	CEDEX 07
departement	Hérault
pays	France

TABLE 8.2 – Exemple d’adresse selon notre modèle

La représentation de l’adresse de l’exemple 1 avec notre modèle est donnée dans le tableau 8.2. Les

coordonnées associées au point sont obtenues en interrogeant l'API ⁷⁸ de l'Etalab, avec comme paramètres des propriétés de l'adresse (informations de la voie, code postal et/ou le nom de la commune).

8.2 Modèle de représentation d'EN *entreprise*

Plusieurs modèles sont proposés dans la littérature pour représenter les EN *entreprise*, presque tous les traitent comme de « simples » POI, avec un focus sur la dimension spatiale. Seul le modèle SIRENE ⁷⁹ représente l'aspect administratif des entreprises. Cependant, il présente quelques limites : (i) il permet seulement d'enregistrer l'activité principale d'une entreprise ; (ii) il permet d'associer une seule adresse à une entreprise : celle du siège social ; (iii) il ne permet pas de représenter les produits et/ou services d'une entreprise.

Face à ces limites, nous avons défini un nouveau modèle de représentation d'EN *entreprise*. La définition de ce nouveau modèle est justifié par le fait que, les entreprises évoluent pendant leur cycle de vie. Elles peuvent développer de nouvelles activités, ouvrir plusieurs filiales et même diversifier leur métier initial. La figure 8.2 illustre notre modèle d'EN *entreprise* à travers un diagramme de classe UML. Nous considérons qu'une EN *entreprise* est composée de deux parties : une partie dédiée à la représentation des informations administratives (données d'immatriculation) et une autre partie qui permet de représenter des données complémentaires, extraites du web.

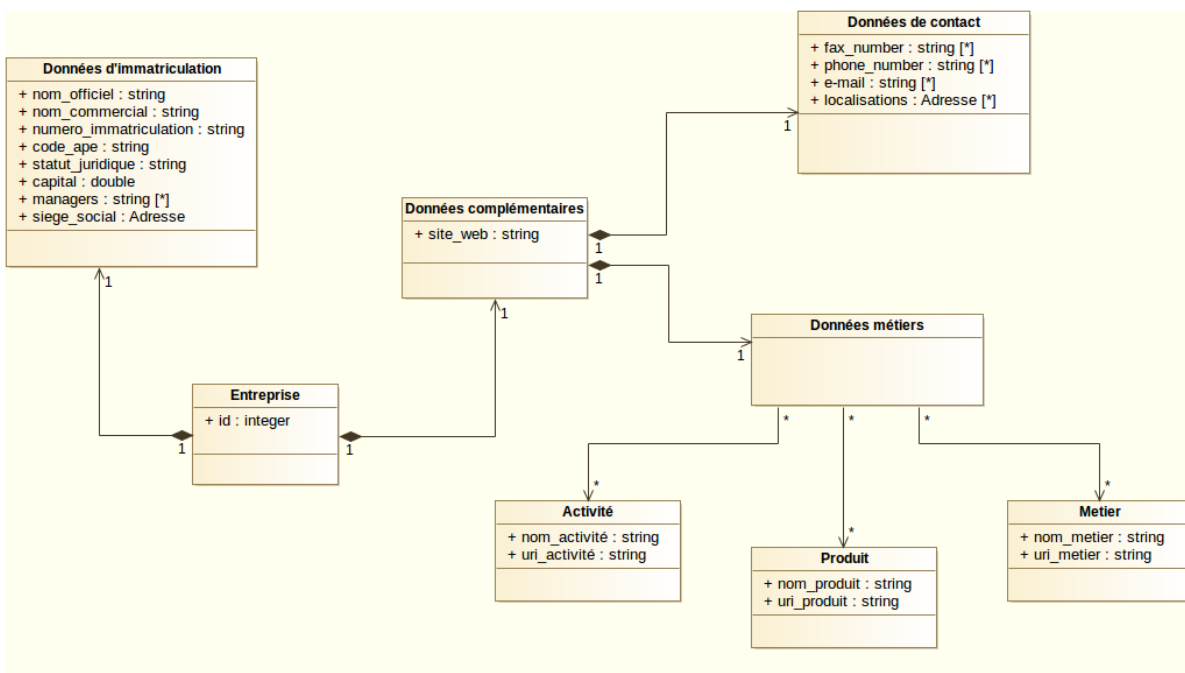


FIGURE 8.2 – Notre modèle de représentation d'une EN *entreprise*

Pour la représentation des données d'immatriculation des EN *entreprise*, nous réutilisons le modèle *SIRENE*. Cette partie de l'EN *entreprise* contient des informations telles que : (i) le nom officiel, qui cor-

78. <https://adresse.data.gouv.fr/api/>

79. <http://www.sirene.fr/sirene/public/static/contenu-base-sirene>

respond au nom d'immatriculation de l'entreprise tel que renseigné dans le registre du commerce ; (ii) le nom commercial, qui est un alias du nom officiel et correspond au nom usuel de l'entreprise. Il peut s'agir notamment du nom d'un des produits phares ou de celui d'une marque ; (iii) le siège social de l'entreprise correspond à une adresse et nous utilisons notre modèle d'adresse pour le représenter ; (iv) le code APE qui correspond à l'activité principale de l'entreprise déclarée lors de son enregistrement.

Les données complémentaires, quant à elles, sont constituées du site web de l'entreprise, ainsi que des données de contact et des données « métiers ». Concernant les données de contact, il s'agit en effet de l'ensemble des informations permettant de rentrer en contact avec l'*entreprise*. Nous avons notamment des numéros de téléphone, numéros de fax, e-mails ou même les différentes localisations. Nous utilisons également notre modèle d'adresse comme un type pour représenter les localisations.

Les données métiers sont constituées de toutes les activités exercées par l'*entreprise*, ses produits et/ou services ainsi que ses métiers. Nous représentons chaque activité (respectivement produit et métier) par son nom et l'URI du concept associé dans la ressource dédiée. En effet, nous avons construit deux ressources basées sur les hiérarchies NAF⁸⁰ (Nomenclature Française des Activités) et CPF⁸¹ (Classification des Produits Française) de l'INSEE. Nous les exploitons pour représenter les activités et produits et/ou services d'entreprises. De même, nous avons construit une ressource dédiée aux métiers exercés au sein des entreprises à partir de la hiérarchie ROME⁸² (Répertoire Opérationnel des Métiers et Emplois) de Pôle Emploi. Cette ressource est utilisée pour représenter les métiers. Le choix de ces trois hiérarchies se justifie par leur niveau de finesse dans la description des différentes catégories. Les propriétés *uri_activite*, *uri_produit* et *uri_metier* du modèle prennent respectivement leurs valeurs dans l'ensemble des URI de concepts des ressources correspondantes.

8.3 Modèle de représentation d'EN *événement*

De nombreux travaux ont porté sur la définition de modèles pour représenter des EN *événement* de type fait et/ou de type socio-culturel. En général, un événement selon ces modèles est défini par un quadruplet $\langle \textit{quoi}, \textit{qui}, \textit{où}, \textit{quand} \rangle$, avec toutes les propriétés au même niveau dans une représentation à plat.

- Le *quoi* indique le thème de l'événement (son titre et/ou sa catégorie).
- Le *qui* correspond aux intervenants de l'événement.
- Le *où* spécifie le lieu de tenue de l'événement.
- Le *quand* donne la période à laquelle se tient l'événement.

Ces modèles considèrent un événement comme quelque chose de ponctuel et qui se déroule dans un lieu bien précis. Cette hypothèse entraîne des redondances pour la représentation de certains types événements. C'est le cas, par exemple, des événements socio-culturels de type tournées ou festivals, qui ont plusieurs prestations dans lesquelles certaines propriétés restent constantes alors que d'autres varient. L'exemple de « la tournée d'Eric Clapton en France » décrit selon différents modèles de état de l'art (section 4.2.5) illustre cette redondance. Chaque prestation de la tournée est un *événement* à part entière, avec les propriétés relatives au *quoi* qui ne changent pas. Des modèles comme SEM [74], EVENT [143] ou Schema.org/Event⁸³, par exemple, permettent de créer des hyper-liens entre les différentes EN, pour

80. <https://www.insee.fr/fr/information/2406147>

81. <https://www.insee.fr/fr/metadonnees/cpfr21/section/A>

82. https://fr.wikipedia.org/wiki/Répertoire_opérationnel_des_métiers_et_des_emplois

83. <http://schema.org/Event>

spécifier qu'il s'agit d'*événements* liés. Ceci est mis en œuvre à travers de la notion de « sous-événement ». Cependant, cette particularité permet de créer une relation entre plusieurs EN sans toutefois résoudre le problème de redondance.

Dans notre contexte, nous nous intéressons uniquement aux événements de type socio-culturels, comme les tournées, pour lesquels les modèles existants impliquent des redondances. Nous proposons donc un modèle qui factorise la partie invariable des EN *événement*, en introduisant la notion de « **représentation** » pour représenter sa partie variable. La figure 8.3 décrit notre modèle dans un diagramme de classe UML. Un événement selon ce modèle comporte une partie « événement » et un ensemble de représentations.

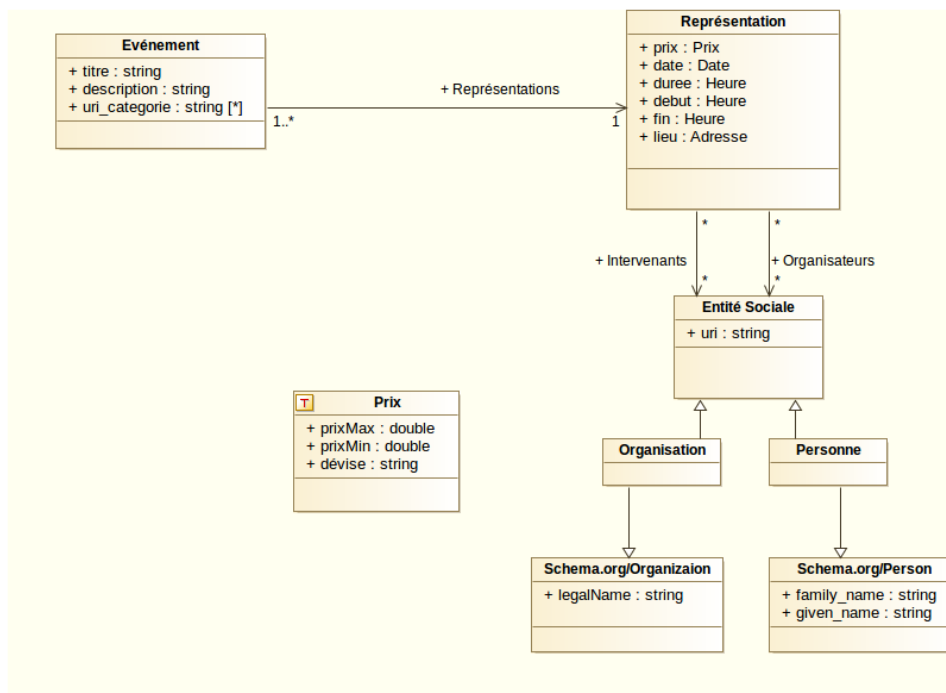


FIGURE 8.3 – Notre modèle de représentation d'une EN *événement*

La partie « événement »

Elle définit le thème (« *quoi* ») de l'*événement* et permet de le caractériser. Elle est exprimée par le titre (le nom de l'*événement*), la ou les catégorie(s) (concert, festival, colloque, etc.) et sa description. Le titre de l'*événement* et sa description sont des chaînes de caractères. En ce qui concerne les catégories d'événements, elles sont représentées par les URI des concepts associés dans une ressource décrivant les catégories d'événements. Nous avons construit ladite ressource en analysant manuellement un ensemble de plateformes en ligne dédiées aux événements. Pour le cas de l'exemple relatif au concert d'Eric Clapton, la partie *événement* de l'EN associée est définie par :

- le titre de l'*événement* : « la tournée d'Eric Clapton en France » ;
- la catégorie : « <http://www.owl-ontologies.com/categoriesEvt.owl#Concert> », URI du concept « concert » dans la ressource ;
- la description : « *Eric Clapton sera en tournée en France en fin d'année. Il se produira à Paris, Lille, Bordeaux et Marseille avec plusieurs artistes invités [...]* » >

Les « représentations »

Une représentation pour un *événement* permet de définir sa partie « variable ». Elle est ponctuelle et est caractérisée par un lieu (*où*), une date (*quand*) et un ensemble d'intervenants et d'organiseurs (*qui*). D'autres propriétés comme les heures de début et fin, la durée ou même le prix des billets sont prises en compte.

Pour exprimer le lieu d'une représentation, nous ré-utilisons le modèle d'adresse défini en 8.1. Les noms des salles (« Bataclan », « Grand Théâtre », etc.) sont stockés dans la propriété *compl_adresse* de l'EN *Adresse*.

Pour représenter la partie temporelle d'une *représentation* (heures de début et de fin, durée), nous utilisons le format ISO-8601⁸⁴.

Les intervenants (acteurs) et les organisateurs d'une *représentation* correspondent à des entités sociales. Pour les représenter, nous utilisons deux classes *Organisation* et *Personne*, construites à partir des classes *Organization* et *Person* de schema.org. Dans ces modèles, nous reprenons les champs permettant de spécifier les noms des personnes ou des organisations. Lorsque l'intervenant est référencé sur le web de données, nous stockons aussi l'URI qui lui est associé.

Enfin, nous représentons les tarifs de billets par un intervalle de prix défini par une valeur inférieure et une valeur supérieure. Ces deux informations sont complétées par la devise dans laquelle sont exprimés ces prix.

8.4 Conclusion

Ce chapitre présente trois modèles que nous avons définis pour représenter respectivement des EN *adresse*, *entreprise* et *événement*. Notre modèle d'EN *adresse* s'inspire des modèles *Adresse* de l'Etalab et *h-adr* de microformats.com. Il permet de représenter un nombre exhaustif d'informations d'adresses, en tenant compte à la fois des aspects spatial et postal de celles-ci. Notre modèle d'EN *entreprise* est basé sur le modèle *SIRENE* que nous enrichissons pour représenter les informations extraites du web, notamment les informations de contact et les informations « métiers » (produits, activités, métiers). En ce qui concerne les EN *événement*, nous proposons un modèle adapté à la représentation d'événements de type socio-culturels. Ce modèle résulte de la factorisation de modèles existants et d'une structuration originale. Il permet de limiter les redondances dans la représentation de certains événements comme les tournées. EN *entreprise* et EN *événement* ré-utilisent le modèle d'EN *adresse* pour représenter leurs informations spatiales.

Ces modèles servent en phase d'extraction d'information pour la spécification des propriétés à identifier dans le texte. Ils sont également utilisés lors de l'indexation pour la structuration des informations extraites. Au chapitre suivant, nous décrivons la chaîne générique que nous avons conçue pour extraire les EN complexes.

84. <https://www.iso.org/fr/iso-8601-date-and-time-format.html>

Chapitre 9

Extraction d'entités nommées complexes dans un texte

Nous avons conçu une architecture générale qui vise l'analyse du contenu textuel du web pour extraire et indexer des EN complexes. Les EN extraites alimentent le module de recherche d'information de notre architecture. Concernant le module d'extraction d'information, nous avons mis au point une chaîne générique qui analyse le texte des pages web pour identifier les EN complexes. Dans ce chapitre, nous présentons cette chaîne en décrivant chacun de ses traitements.

La première section de ce chapitre rappelle notre problème et les différents verrous traités. La deuxième section décrit en détail le fonctionnement de notre chaîne générique. Les sections suivantes présentent l'instanciation et l'expérimentation de cette chaîne générique pour l'extraction de trois types d'EN complexes : les EN *adresse*, *entreprise* et *événement*.

9.1 Rappel du problème et contexte

L'objectif du module d'extraction d'information est de traiter la collection de textes obtenue en sortie du module des prétraitements, pour en extraire des EN complexes. Dans notre contexte, plusieurs verrous sont à prendre en considération.

- Le premier verrou correspond à l'extraction d'information dans un contexte bruité. Pour illustrer ce verrou, considérons le texte de la figure 9.1 qui est extrait d'une page décrivant un événement. Les syntagmes mis en évidence par les rectangles en traits continus correspondent à des propriétés de l'événement décrit dans le texte, tandis que ceux dans les rectangles en pointillés correspondent à du bruit. En effet, l'événement traité dans cette page s'intitule « *Jeff Panacloc contre-attaque* ». Mais la description de celui-ci fait référence au titre d'un spectacle précédent (« *Jeff Panacloc perd le contrôle* ») du même intervenant, ainsi qu'à la date de la dernière représentation de celui-ci (« *10 Avril 2016* »). De plus, les noms des membres de l'équipe technique (le « *metteur en scène* » et le « *metteur en lumière* »), tout comme les noms des personnages (« *Jeff Panacloc* » et « *Jean-Marc* ») sont également mentionnés dans ce texte. Tous ces noms correspondent à des noms de personnes et seuls ceux qui renvoient aux acteurs de l'événement nous intéressent. L'enjeu pour nous est de faire la distinction entre toutes ces informations pour extraire uniquement celles qui

correspondent à des propriétés de l'événement décrit.

ticketmaster.com

Billetterie show **JEFF PANACLOC CONTRE-ATTAQUE** ticketmaster

HUMOUR ET ONE (WO)MAN SHOW - HUMOUR

Après leur premier spectacle, "Jeff Panacloc perd le contrôle", un burn out, une thalasso, et plusieurs délits de fuite, après une tournée triomphale de trois ans couronnée par la prestation parisienne du 10 Avril 2016 dernier

Jeff Panacloc et **Jean-Marc** reviennent dans de nouvelles aventures. Ils se produisent à la **SALLE EURYTHMIE, Rue Salvador Allende 82000 MONTAUBAN**, ce **samedi 14 octobre 2017**. Le prix des places est compris entre : 32.00 et 44.50 €

La famille s'agrandit avec de nouveaux personnages aussi déjantés les uns que les autres !
 Jean-Marc supportera-t'il cette nouvelle cohabitation ?
 Jeff reprendra-t'il enfin le contrôle ?
 Toutes les réponses ou presque dans ce nouveau spectacle

Metteur en scène : **Nicolás NEBOT**
 Mise en Lumières : **Erwan CHAMPIGNE**

Réservez vos places à partir du **06 Septembre 2017**

FIGURE 9.1 – Texte extrait d'une page décrivant un événement

- Le deuxième verrou correspond à la détection d'information dont les formes ne sont pas régulières. De plus, les syntagmes correspondant sont généralement noyés dans le texte sans marquage particulier, ce qui rend leur identification difficile. Considérons l'exemple du texte de la page d'accueil d'un site web d'entreprise illustré par la figure 9.2. Les rectangles en traits continus mettent en évidence des syntagmes qui relèvent du champ lexical de l'activité de l'entreprise (travaux de charpente). Ces syntagmes correspondent à des chaînes de caractères (mots ou groupes de mots) qui sont complètement noyées dans le texte de la page. De même, certaines propriétés d'événements présentent aussi cette caractéristique. Le titre, par exemple, peut correspondre à un mot (« *Soprano* »), à un groupe de mots (« *Jeff Panacloc contre-attaque* »), ou même à toute une phrase (« *Romanès cirque tsigane : les Nomades tracent les chemins du ciel!* »).
- Le troisième verrou correspond au traitement de la sur-composition dans un processus d'extraction d'information. Pour le cas de l'événement de la figure 9.1, le titre par exemple, contient le nom d'un intervenant. De même, le lieu (donné ici par une adresse) contient le nom de la salle (« *Salle Eurythme* »), et le nom de la commune (« *Montauban* ») qui sont eux mêmes des EN de type *lieu*. L'enjeu ici est d'identifier sans ambiguïté et dans sa globalité le syntagme correspondant à la propriété traitée.
- Le dernier verrou correspond à la généralité du processus d'extraction d'EN complexes à mettre sur pied.

Nous avons donc conçu une chaîne générique d'extraction d'EN complexes dans le texte. Celle-ci ne dépend pas du type d'EN traité, mais uniquement du modèle les décrivant et du contexte dans lequel se fait leur extraction.



FIGURE 9.2 – Texte extrait de la page d'un site web d'entreprise

9.2 Proposition d'une chaîne générique d'extraction d'EN complexes dans un texte

Notre chaîne générique d'extraction d'EN complexes, illustrée par la figure 9.3, est composée de trois modules de traitement. Elle prend en entrée le modèle décrivant l'EN complexe ciblée, la liste des propriétés bruitées et un texte à traiter. Le texte est analysé pour identifier les propriétés de l'EN complexe avant de les agréger. Le premier module détecte les blocs susceptibles de contenir des propriétés bruitées, le deuxième module, quant à lui, analyse chaque bloc pour identifier les propriétés qu'il contient et le dernier module regroupe les annotations de propriétés pour construire des instances d'EN complexes.

Détection de blocs correspondant à des propriétés

Ce premier module vise particulièrement le verrou lié à l'extraction d'EN complexes dans un contexte bruité. Son objectif est de détecter les blocs de texte contenant des propriétés bruitées afin de les annoter correctement par la suite. Reprenons l'exemple du texte décrivant l'événement intitulé « *Jeff Panacloc contre-attaque* » (voir figure 9.1). La mise en œuvre de la détection de blocs sur ce texte est illustrée par la figure 9.4. Les blocs contenant les propriétés bruitées sont représentés sur la figure par des rectangles en traits continus. Ce traitement permet de mettre en évidence la date de l'événement dans le « *bloc représentation* ». Ce qui permet, par ailleurs, de faire la distinction avec la date de l'autre événement évoqué dans la page, tout comme celle de mise en vente des billets (encadrées par un trait discontinu dans la figure 9.1).

La mise en œuvre de ce module lors de l'instanciation de la chaîne générique s'appuie sur deux approches : l'apprentissage ou les patrons. En effet, lorsque le bloc contenant la propriété bruitée suit une forme régulière dans le corpus analysé, les patrons sont plus adaptés. Dans le cas contraire, nous préconisons l'utilisation de l'apprentissage supervisé pour le marquage des blocs.

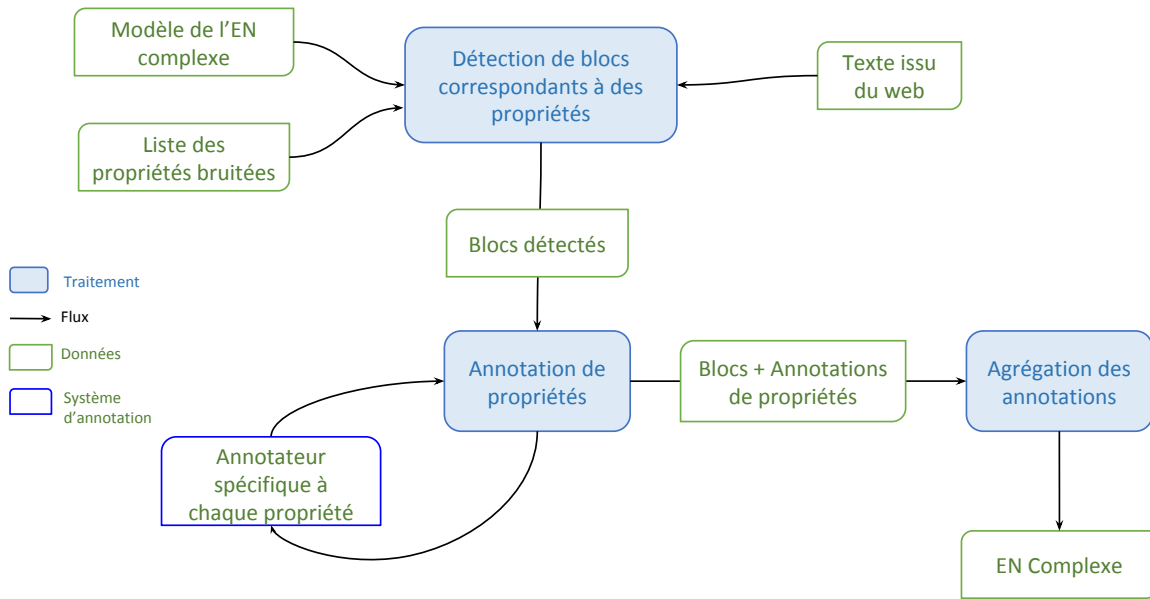


FIGURE 9.3 – Chaîne générique d'extraction d'EN complexes dans un texte

Les blocs obtenus au terme de ce premier traitement peuvent ensuite être analysés pour l'identification des propriétés qu'ils contiennent.

Bloc titre → **ticketmaster.com**

Billetterie show JEFF PANACLOC CONTRE-ATTAQUE ticketmaster

Genre HUMOUR ET ONE (WO)MAN SHOW - HUMOUR ← *Bloc catégorie*

Après leur premier spectacle, "Jeff Panaclor perd le contrôle", un burn out, une thalasso, et plusieurs délits de fuite, après une tournée triomphale de trois ans couronnée par la prestation parisienne du 10 Avril 2016 dernier

Jeff Panaclor et Jean-Marc reviennent dans de nouvelles aventures. Ils se produisent à la SALLE EURYTHMIE, Rue Salvador Allende 82000 MONTAUBAN, ce samedi 14 octobre 2017. Le prix des places est compris entre : 32.00 et 44.50 €

La famille s'agrandit avec de nouveaux personnages aussi déjantés les uns que les autres !
 Jean-Marc supportera-t'il cette nouvelle cohabitation ?
 Jeff reprendra-t'il enfin le contrôle ?
 Toutes les réponses ou presque dans ce nouveau spectacle

Metteur en scène : Nicolas NÉBOT
 Mise en Lumières : Erwan CHAMPIGNE

Réservez vos places à partir du 06 Septembre 2017

Bloc représentation →

FIGURE 9.4 – Exemple de texte, décrivant un événement et contenant des propriétés bruitées

Annotation de propriétés

Ce module est dédié à l'identification des propriétés d'EN complexes dans les blocs détectés par le premier module. Les propriétés non bruitées sont annotées en analysant tout le texte comme un seul bloc. L'annotateur, le plus adapté à chaque type de propriété, est mis en œuvre pour ce traitement (cf. section 5.2). Les différents systèmes d'annotations présentés dans l'état de l'art peuvent être utilisés pour l'annotation des propriétés dans les blocs. De même, des annotateurs personnalisés peuvent être conçus en fonction du corpus traité ou des spécificités de la propriété ciblée, puis intégrés dans la chaîne.

La figure 9.5 illustre la mise en œuvre de l'annotation des propriétés dans le texte décrivant l'événement correspondant au spectacle « *Jeff Panacloc contre-attaque* ». Sur le « *bloc titre* », nous invoquons l'annotateur dédiée, sur le « *bloc représentation* » de la figure 9.4, nous invoquons tour à tour un annotateur d'adresses, un annotateur de dates, un annotateur d'intervenants et un annotateur de prix. Toutes les annotations de propriétés obtenues en analysant les blocs sont mises en évidence sur la figure 9.5 par les rectangles bleus en traits pointillés fins. En sortie de ce module, nous avons les blocs détectés et les annotations de propriétés qu'ils contiennent.

ticketmaster.com

titre

Billetterie show: JEFF PANACLOC CONTRE-ATTAQUE ticketmaster

Genre: HUMOUR ET ONE (WO)MAN SHOW - HUMOUR catégorie

Après leur premier spectacle "Jeff Panacloc perd le contrôle", un burn out, une thalasso, et plusieurs délits de fuite, après une tournée triomphale de trois ans couronnée par la prestation parisienne du 10 Avril 2016 dernier

intervenants

Jeff Panacloc et Jean-Marc reviennent dans de nouvelles aventures. Ils se produisent à la SALLE EURYTHMIE, Rue Salvador Allende 82000 MONTAUBAN, ce samedi 14 octobre 2017. Le prix des places est compris entre : 32.00 et 44.50 €

lieu

prix

date

La famille s'agrandit avec de nouveaux personnages aussi déjantés les uns que les autres !
 Jean-Marc supportera-t'il cette nouvelle cohabitation ?
 Jeff reprendra-t'il enfin le contrôle ?
 Toutes les réponses ou presque dans ce nouveau spectacle

Metteur en scène : Nicolas NEBOT
 Mise en Lumières : Erwan CHAMPIGNE

Réservez vos places à partir du 06 Septembre 2017

FIGURE 9.5 – Exemple de texte, décrivant un événement avec les propriétés annotées

Agrégation des annotations

Une fois les propriétés identifiées, ce dernier module a pour vocation de les agréger en instances d'EN complexes. Deux scénarios sont à distinguer selon que le texte contient une seule instance d'EN complexes ou plusieurs. Si le texte contient une seule instance d'EN (comme l'exemple de la figure 9.5 qui décrit un seul événement), l'agrégation consiste simplement à associer chaque propriété annotée au champ correspondant dans le modèle.

Dans le cas où le texte contient plusieurs instances d'EN complexes, nous préconisons l'utilisation des patrons ou d'algorithmes « dédiés ». Le choix de l'approche dépend des spécificités du texte. Lorsque dans un texte les propriétés d'une instance d'EN complexe se suivent, l'approche d'agrégation basée sur les patrons nous semble la plus adaptée ([128], [3]). Dans le cas où les propriétés d'instances distinctes d'EN sont éparpillées dans le texte et/ou « s'entrecoupent » (corpus traité par Foley et al. [55] par exemple), nous recommandons l'utilisation d'algorithmes « dédiés » pour l'agrégation de propriétés.

Dans les sections suivantes, nous instancions notre chaîne générique pour extraire dans différents contextes, trois types d'EN complexes : les EN *adresse*, les EN *entreprise* et les EN *événement*.

9.3 Extraction d'EN adresse

Dans les modèles de représentation des EN *entreprise* et *événement*, nous utilisons le modèle d'EN *adresse* pour représenter respectivement les localisations d'entreprises et les lieux de représentations d'événements. Ces *adresses* sont traitées dans notre contexte comme des EN complexes (cf. section 8.1). Nous instancions de ce fait notre chaîne générique pour les extraire.

9.3.1 Contexte d'extraction d'EN *adresse*

Les adresses traitées dans le cadre de notre projet sont contenues sur des sites web d'entreprises ou des plateformes dédiées aux événements. Pour ces deux types de corpus, nous avons constaté que les propriétés d'une adresse sont généralement regroupées en syntagmes qui se suivent. Par conséquent, aucune propriété n'est bruitée, puisque chacune n'a de sens que lorsqu'elle est située à côté des autres.

Par ailleurs, il existe une norme (la norme *AFNOR NF Z 10-001*⁸⁵) qui définit les règles d'écriture d'adresses françaises. Une adresse selon la norme *AFNOR NF Z 10-001* doit contenir dans l'ordre : les informations complémentaires (étage, entrée, bâtiment, bloc, ...), les informations de la voie (numéro, extension de numéro et nom de la voie), les informations particulières de distribution (boîte postale, course spéciale ...), le code postal, la commune de distribution et les informations spéciales de distribution (Cedex, Cidex). Le listing 9.1 illustre un exemple d'adresse conforme à la norme AFNOR NF Z 10-001.

```
1 Résidence Clé des Champs
2 7 Rue Saint John Perse
3 BP 1161
4 64000 Pau Cedex
5
```

Listing 9.1 – Exemple d'adresse suivant la norme AFNOR NF Z 10-001

Cependant dans la pratique, ces règles sont très peu respectées et plus particulièrement sur le web. Dans la majorité des cas, les syntagmes correspondant aux différentes propriétés sont permutés. De ce fait, le problème auquel nous nous intéressons porte sur l'identification des différentes propriétés en tenant compte de la permutation éventuelle de celles-ci. Le listing 9.2 illustre un exemple d'adresse pour laquelle

85. <http://www.76310.fr/afnor-xpz.htm>

les règles de la norme ne sont pas respectées. En effet, cette adresse comporte plusieurs compléments d'adresses (ligne 1 et ligne 3). À la ligne 3, un complément d'adresse lié au « bâtiment » (*Bat A*) suit le nom de la voie, et un autre suit le numéro de boîte postale (*4ème Etg*). Nous constatons aussi que le code postal et le nom de la commune sont inversés. De plus, plusieurs abréviations sont utilisées dans cet exemple : Z.I. pour zone industrielle, Bat pour bâtiment, Etg pour étage.

```

1   Z.I. du Phare - Mérignac
2   3, Impasse Rudolf Diesel,
3   Bat A - BP 50227, 4ème Etg,
4   Mérignac, F-33708 Cedex, France.
5

```

Listing 9.2 – Exemple d'adresse ne respectant pas la norme AFNOR NF Z 10-001

Certaines propriétés d'adresses comme le code postal, le nom de la commune ou le numéro de boîte postale peuvent être identifiées simplement en exploitant des expressions régulières ou des dictionnaires. Dans notre contexte, les principales difficultés se situent au niveau de l'extraction de deux propriétés : le complément d'adresse et le nom de la voie.

- Concernant le complément d'adresses, nous relevons trois difficultés majeures : (i) il peut se retrouver au début de l'adresse ou même en plein milieu comme l'exemple du listing 9.2 ; (ii) il peut être sur-composé (*Z.I. du Phare - Merignac*) ; (iii) plusieurs compléments d'adresse peuvent figurer dans l'adresse sans se suivre pour autant (voir exemple du listing 9.2, ligne 3).
- Concernant le nom de la voie, nous avons deux difficultés : (i) un nom de voie est généralement introduit dans une adresse française par un mot clé (*boulevard, rue, etc.*), cependant, il est difficile de savoir où il se termine. En effet, le nom de voie peut contenir plusieurs mots après l'introducteur et comme on ne sait pas a priori quelle est la propriété qui le suit (conséquence du non respect de la norme) il est difficile d'identifier sa fin. Contrairement à Alhers et al. [3], nous ne disposons pas en libre accès, d'un dictionnaire avec tous les noms de voie en France ; (ii) un nom de voie peut être sur-composé et contenir notamment un nom de commune (*21 Rue de Nancy Paris 75010*), la difficulté dans ce cas est de l'extraire sans confondre la commune de l'adresse à celle contenue dans le nom de voie lorsqu'elles se suivent directement dans l'adresse.

9.3.2 Mise en œuvre de la chaîne générique pour l'extraction d'EN *adresse*

Nousinstancions la chaîne générique de la figure 9.3 pour extraire des EN *adresse*. Notre contexte d'extraction n'est pas bruité, de ce fait, le texte est traité comme un seul bloc. Nous présentons l'annotation des propriétés puis leur agrégation.

Annotation de propriétés d'EN *adresse*

Comme les propriétés d'une adresse sont regroupées dans les textes analysés, nous avons observé les différentes formes d'adresses utilisées à partir d'un échantillon représentatif de 150 sites web. Ces formes sont exploitées pour identifier et pour valider les différentes propriétés d'une adresse. Dans la grande majorité des adresses que nous avons observées (95% des adresses dans les 150 sites web), le code postal était présent. C'est la raison pour laquelle nous ne traitons que les adresses contenant un code

postal. Notre approche d’annotation de propriétés est basée sur des patrons et elle exploite également des dictionnaires. Le tableau 9.1 donne la notation permettant de décrire les patrons.

Notations	Significations
$[N_1 - N_2]$	un élément entre N_1 et N_2
$A?$	l’élément A apparaît 0 ou 1 fois
A^*	A est apparaît 0 ou plusieurs fois
$A B$	A est directement suivi de B
$A B$	A ou B
$A[n, m]$	A répété entre n et m fois

TABLE 9.1 – Règles et symboles de notations pour les patrons

Le tableau 9.2, quant à lui, définit la notation de propriétés utilisées dans l’écriture des patrons.

Sens	Notations	Exemples
Complément d’adresse	CA	Résidence Rigaud
Introducteur de CA	ICA	Résidence
Code postal	BP	BP 1167
Course spéciale	CS	CS 2587
Numéro de voie	NV	10
Indice de répétition du NV	R	ter
Type de voie	TNV _o	Avenue
Nom de la voie	NV _o	Avenue de l’université
Code postal	CP	64000
Nom de la commune	C	Pau
Numéro de courrier	NC	Cedec 01
Département	D	Pyrénées-Atlantiques
Pays	P	France
Un mot quelconque du texte	Token	république

TABLE 9.2 – Notations des propriétés d’adresse

Les formes observées sur les 150 sites web ont été regroupées en trois grandes catégories, ceci en fonction de la position des propriétés renseignées.

- **Catégorie 1** : le nom de la voie (NV_o) et le code postal (CP) sont requis et il n’y a pas de complément d’adresse entre eux (Z.I Les Cailloux, 14 Rue de la Feculerie, 45150 - JARGEAU). L’adresse est définie dans ce cas par la forme suivante :

$$(1) \text{ Adresse} \rightarrow CA^* ((BP \ CS)|(CS \ BP)|BP|CS)? \\ NV? \ R? \ NV_o \ ((BP \ CS)|(CS \ BP))? \\ (CP \ C?)|(C? \ CP) \ NC? \ D? \ P$$

- **Catégorie 2** : le nom de la voie (NV_o), un complément d’adresse (CA) et le code postal (CP)

sont requis ; le complément d'adresse étant entre le nom de la voie et le code postal (90, avenue de Canejan - Parc Industriel 33600 - PESSAC). La forme résumant les adresses de cette catégorie est la suivante :

$$(2) \text{ Adresse} \rightarrow CA * ((BP \ CS)|(CS \ BP)|BP|CS)? \\ NV? \ R? \ NVo \ ((BP \ CS)|(CS \ BP))? \\ CA \ (CP \ C?)|(C? \ CP) \ CA * \ NC? \ D? \ P$$

- **Catégorie 3** : le code postal et le nom de la commune sont requis, mais il n'y a pas de nom de voie (Actipôle 85 - 85170 Belleville-sur-vie). Dans ce cas, la forme de l'adresse est donnée par :

$$(3) \text{ Adresse} \rightarrow CA * ((BP \ CS)|(CS \ BP)|BP|CS)? \\ (CP \ C?)|(C? \ CP) \ NC? \ D? \ P$$

C'est à partir de ces trois principales formes que nous avons mis au point les patrons dédiés aux annotations des propriétés d'adresses. Ces annotations se font dans un ordre bien défini, puisque certaines sont nécessaires pour l'annotation des autres. Notre contribution se situe au niveau de l'annotation du complément d'adresse et du nom de la voie. Ce sont ces processus d'annotation que nous détaillerons par la suite.

Annotation du complément d'adresse (CA) - Les compléments d'adresses sont introduits par des mots clés particuliers (bâtiment, résidence, zone industrielle, zénith, stade, théâtre, technopole, etc.). Nous avons répertorié tous ces mots clés ainsi que leurs différentes abréviations dans un dictionnaire. L'analyse de l'échantillon des 150 sites web révèle qu'un complément d'adresse comporte habituellement, en plus de son introducteur, entre 1 et 6 autres mots. Le patron dédié à son identification dans le texte est le suivant :

$$CA \rightarrow ICA \ Token[1, 6] \quad (4)$$

Le déclenchement de ce patron dépend de la position du complément d'adresse.

- Si le complément d'adresse est au début d'une adresse de la catégorie 1, le patron ne se déclenche que, si en plus de l'introducteur du complément d'adresse (ICA), l'une au moins des informations suivantes a été annotée (dans cet ordre) : un autre introducteur de complément d'adresse (ICA), un numéro de boîte postale (BP), un numéro de course spéciale (CS), une numéro de voie (NV) ou un type de voie (TNVo).

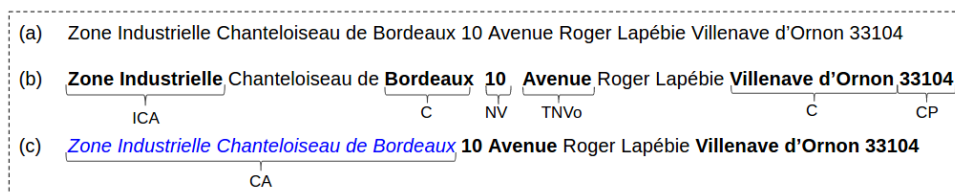


FIGURE 9.6 – Exemple d'instanciation du patron (4) pour annoter un CA en début d'adresse

La figure 9.6 illustre un exemple d'instanciation du patron (4). Le texte de l'adresse correspond à la ligne (a). Une première passe annotera d'abord les introducteurs de noms de voie et de complément

d’adresse, tout comme le numéro de la voie, les noms de commune et le code postal (ligne (b)). On a bien une séquence de moins de 6 mots (*Chanteloise de Bordeaux*) entre l’introducteur de complément d’adresse (*ICA*) et le numéro de la voie (*NV*) : ce qui déclenche le patron (4).

- Le complément d’adresse suit le nom de la voie (catégorie 2) ou précède le code postal dans une adresse sans voie (catégorie 3). Dans ce cas, le patron (4) ne se déclenche que, si après l’introducteur du complément d’adresse (*ICA*), on a détecté au moins (dans cet ordre) : un autre introducteur de complément d’adresse (*ICA*), un numéro de boîte postale (*BP*), un numéro de course spéciale (*CS*), un code postal (*CP*).

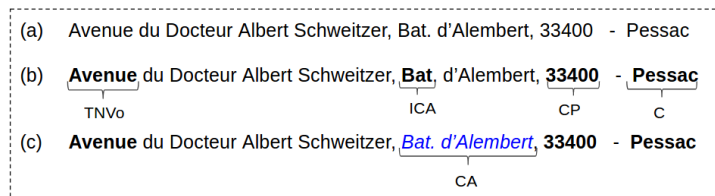


FIGURE 9.7 – Exemple d’instanciation du patron (4) pour annoter un CA en milieu d’adresse

la figure 9.7 illustre l’instanciation du patron (4) pour annoter un complément d’adresse situé en milieu d’adresse. La première phase d’annotation ici fonctionne exactement comme pour le cas précédent. La ligne (b) montre bien qu’on a identifié un code postal après l’introducteur du complément d’adresse avec moins de 6 mots entre eux. Notre processus annotera donc le syntagme « *Bat. d’Alembert* ».

Annotation du nom de la voie (*NVo*) - Tout comme les compléments d’adresse, les noms de voie sont introduits par des mots clés correspondant aux types de voie (boulevard, avenue, rue, etc.). Nous avons constitué un dictionnaire avec ces différents mots clés et leurs abréviations. Le problème pour cette propriété étant la détection que sa fin, nous avons observé l’échantillon des 150 sites web pour évaluer le nombre de syntagmes contenus dans un nom de voie. Cette observation a permis d’en déduire que le nom de la voie est généralement constitué de 8 mots au maximum. De ce fait, nous utilisons le patron suivant pour l’annoter.

$$NVo \rightarrow TNVo \quad Token[1, 7] \quad (6)$$

Ce patron est déclenché de la même façon pour les adresses des catégories 1 et 2. L’identification d’un type de voie et d’au moins, l’un des éléments suivants suffit à déclencher le patron : un introducteur de complément d’adresse (*ICA*), un numéro de boîte postale (*BP*), un numéro de course spéciale (*CS*), un code postal (*CP*) ou un nom de commune (*C*). La figure 9.8 illustre une instanciation du patron (6) pour annoter le nom de la voie dans l’adresse : *2 Avenue du Président Pierre Angot, Technopole Hélioparc - Bat. De Vinci, 64 000 - Pau*. Ici, c’est l’introducteur du complément d’adresse (*Technopole*) qui marque la fin de la voie.

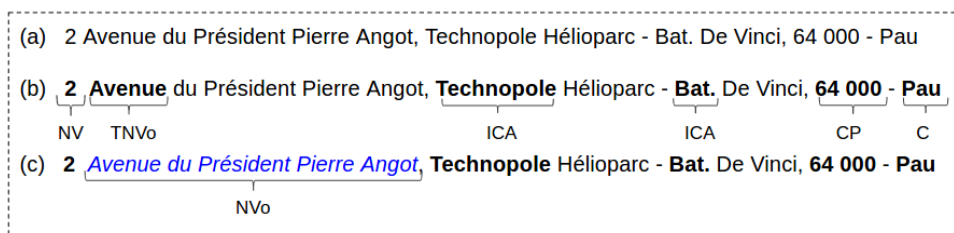


FIGURE 9.8 – Instanciation du patron (6) pour annoter un nom de voie

Un problème particulier se pose dans le cas où le nom de la voie se termine par un nom de commune et que la commune de l'adresse suivent directement ce nom de voie. Pour éviter toute confusion entre la commune de l'adresse et celle de la voie, nous exploitons la position du code postal pour annoter sans erreur le nom de la voie. Pour l'adresse de la figure 9.9 (*1 Route de Toulouse Bordeaux 33 000*), le fait d'annoter « *Bordeaux* » juste avant le code postal et qu'un autre nom de commune n'ait pas été annoté après ce dernier, permet d'inclure « *Toulouse* » dans le nom de la voie.

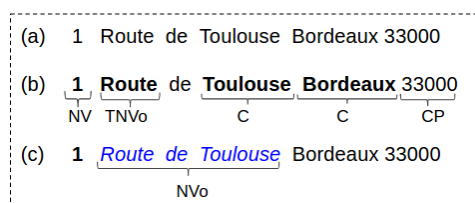


FIGURE 9.9 – Instanciation du patron (6) pour annoter un nom de voie se terminant par un nom de commune

Dans le cas où un complément d'adresse se termine par un nom de commune et se trouve entre le nom de la voie et la commune de l'adresse (*10 Avenue Roger Lapébie, Zone Industrielle Chanteloise de Bordeaux, Villenave d'Ornon 33104*), nous avons exactement le même problème que celui soulevé précédemment (confusion de la commune de l'adresse avec celle du complément d'adresse). Pour le résoudre, nous exploitons, comme pour le cas du nom de la voie, la position du code postal.

Agrégation des propriétés d'EN *adresse*

Nous avons vu en présentation de notre contexte que les propriétés d'une adresse dans un texte se suivent. De ce fait, nous utilisons une approche par patrons pour agréger les annotations de propriétés d'adresses. Nous ré-utilisons les principales formes ((1), (2) et (3)) définies précédemment pour écrire les patrons d'agrégation. Dans chacun de ces trois patrons, le code postal est l'élément déclencheur et il est requis pour toutes les adresses traitées.

Pour le cas des textes contenant plusieurs adresses (une entreprise avec plusieurs sites d'exploitation par exemple), le fait que les propriétés d'adresses se suivent dans le texte permet d'agréger les différentes instances.

9.3.3 Expérimentation de l’extraction d’EN *adresse*

Nous expérimentons l’annotateur d’adresses obtenu en mettant en œuvre la chaîne générique décrite dans la section précédente. Nous évaluons ici la capacité de cet annotateur à identifier correctement une adresse dans un texte. Le corpus d’évaluation est constitué de 250 textes issus de pages web.

9.3.3.1 Protocole d’évaluation de l’annotation d’EN *adresse*

Il n’existe pas, à notre connaissance, de campagnes d’évaluation qui ciblent les EN de type adresse. Nous en avons donc construit une, en nous inspirant des campagnes TREC [179] pour l’évaluation des systèmes de RI. Dans cette campagne :

- les « catégories » correspondent aux types d’EN à annoter : les adresses dans notre cas ;
- les « qrels » (*query relevance judgements ou jugements de pertinences*) correspondent ici aux adresses pertinentes de chaque texte selon l’expert.

Pour l’ensemble du corpus d’évaluation, les adresses extraites par notre système et les « qrels » sont comparés afin de calculer la précision et le rappel pour chaque texte. La précision globale sur l’ensemble du corpus est obtenue en faisant la moyenne arithmétique des précisions. Il en est de même du rappel global. La moyenne harmonique du rappel et de la précision globales donne la F_1 -mesure.

Les adresses annotées partiellement sont considérées incorrectes, il en est de même pour celles avec des propriétés mal annotées.

9.3.3.2 Analyse des résultats de l’annotation d’EN *adresse*

Nous avons expérimenté notre système d’annotation d’adresses sur l’ensemble des textes du corpus d’évaluation. Le tableau 9.3 présente les résultats obtenus. Il comporte également les résultats suivant les trois catégories observées (voir section 9.3.2).

Catégories	Précision en %	Rappel en %	F_1 -Mesure en %
Tout	80,00	81,00	80,82
catégorie 1	88,00	89,00	88,33
catégorie 2	74,00	77,00	75,55
catégorie 3	93,00	93,00	93,00

TABLE 9.3 – Résultat de l’évaluation de l’annotation d’adresses

Sur l’ensemble du corpus d’évaluation, le rappel obtenu est de 81%. Ce qui est à peu près la moyenne obtenue pour les systèmes d’extraction d’adresses basés sur des patrons ([162], [106]). La précision globale de notre système est de 80%. Ce résultat est particulièrement intéressant dans le sens où d’autres systèmes ciblant l’extraction d’adresses en tenant compte d’informations complémentaires, comme celui conçu par Schmidt et al. [162], ne réalise qu’une précision de 65%. Dans leur contexte, la détection des informations complémentaires est la principale cause d’annotations erronées. Les différentes formes prises en compte par nos patrons permettent de limiter la proportion d’annotations incorrectes.

Pour les adresses de la catégorie 1, nous constatons que notre système les identifie correctement dans presque 90% des cas. Les quelques adresses qui ne sont pas annotées correspondent à des adresses sans codes postaux (10 Place de la Bourse, Bordeaux) et/ou sans nom de commune (10 Place de la Bourse).

Les sites web contenant ce type d'adresse sont très rares dans nos corpus d'extraction d'EN *entreprise* et *événement* (moins de 5%). Par ailleurs, la prise en compte de ces formes particulières introduisait beaucoup de bruit dans l'annotation du nom de voie et du complément d'adresse ; c'est pourquoi nous ne les avons pas traitées. Le silence observé est également dû à des erreurs de traitements du texte (échec de l'analyse morpho-syntaxique dû à la perte de la structure lors de la transformation en texte du contenu des pages).

Concernant les adresses pour lesquelles un complément d'adresse est situé entre le nom de la voie et le code postal (catégorie 2), notre système permet d'en extraire correctement un grand nombre (F_1 -score autour de 75%). Par contre, les adresses extraites ont parfois des propriétés mal annotées, notamment le nom de la voie. Les erreurs d'annotation du nom de voie surviennent en général lorsque le complément d'adresse n'a pas été identifié correctement et le système l'inclut à tort dans le nom de la voie.

Pour les adresses spécifiées uniquement par le nom de la commune et le code postal, les résultats obtenus par notre système sont très bons (précision et rappel autour de 93%). Ces résultats sont du même ordre que ceux obtenus par Nesi et al. [128], qui pourtant travaillent sur l'annotation de formes d'adresses n'intégrant pas les compléments d'adresses.

9.4 Extraction d'EN *entreprise*

Le module d'extraction d'information du service *Cognisearch Business* s'appuie sur notre chaîne générique. Nous précisons qu'un prétraitement est mis en œuvre en amont pour filtrer le web et identifier pour chaque entreprise son site web (ce prétraitement est décrit au chapitre 11, section 11.1.1). Notre chaîne générique est ensuite mise en œuvre pour analyser le texte de ces sites web afin d'extraire les données « métiers » et les celles de contact. Nous présentons dans la suite, les spécificités du corpus analysé, la mise en œuvre de la chaîne générique ainsi que son expérimentation.

9.4.1 Contexte d'extraction d'EN *entreprise*

Le texte traité pour l'extraction des informations d'une entreprise est issu des pages de son site web. Nous formulons l'hypothèse suivante : « *on ne trouve pas dans les pages du site web d'une entreprise des informations concernant d'autres entreprises* ». Cette hypothèse est légitime dans la mesure où le site web d'une entreprise constitue sa principale vitrine sur le web et donc, elle ne saurait y mettre en avant les informations d'une autre entreprise. Par conséquent, nous assumons que les propriétés ne sont pas bruitées dans les textes que nous analysons.

Cette extraction pose par ailleurs deux principales difficultés : (i) les informations « métiers » (activités, métiers, produits) d'entreprises sont noyées dans le texte des pages et celles-ci ne suivent pas de formes régulières (voir figure 9.2) ; (ii) les adresses d'entreprise sur leur site web ne sont pas toujours exprimées suivant des standards. En plus du code postal, du nom de la commune et des informations de la voie, des compléments d'adresses sont également renseignés pour en préciser la localisation.

9.4.2 Mise en œuvre de la chaîne générique pour l'extraction d'EN *entreprise*

La chaîne générique de la figure 9.3 est instanciée pour l'extraction d'EN *entreprise*. Les propriétés n'étant pas bruitées, tout le texte est traité comme un seul bloc.

Annotation des propriétés d'EN *entreprise*

Ce module analyse le texte des pages du site web de chaque entreprise pour en extraire des informations « métiers » et de contact. La figure 9.10 illustre ce processus d'extraction. Selon les propriétés traitées, des approches d'extraction basées sur des ressources de type connaissance et celles exploitant des patrons sont mises en œuvre.

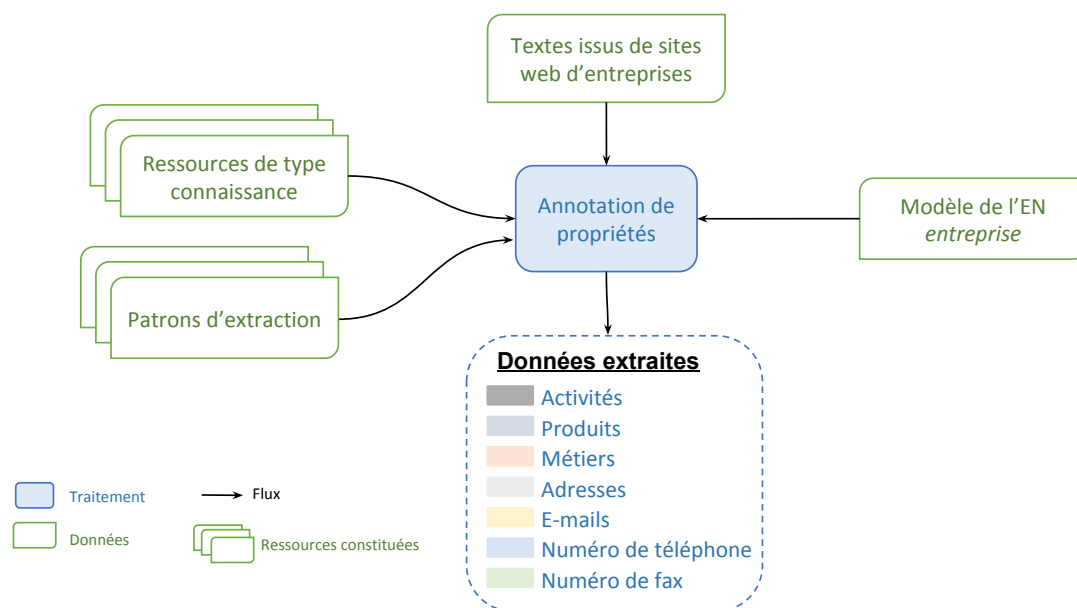


FIGURE 9.10 – Annotation de propriétés d'EN *entreprise*

Annotation des informations « métiers » - Nous exploitons pour cette tâche une approche basée sur des ressources de type connaissance. En effet, trois ressources relatives aux activités, aux produits et aux services ainsi qu'aux métiers sont projetées sur le texte afin d'identifier des instances de concepts. La ressource des activités, par exemple, est représentée par une structure arborescente de 5 niveaux hiérarchiques, elle est constituée de 1700 concepts pour 3400 labels. Les labels caractérisant chaque concept sont tokenisés, lemmatisés et normalisés. Nous appliquons quelques transformations sur le texte avant de traiter ses mots de la même façon que les labels de la ressource. Ces transformations visent notamment la décomposition de certains syntagmes pour plus d'efficacité lors de la projection de la ressource. Le syntagme « *charpentes et couvertures traditionnelles* », par exemple, sera transformé en « *charpentes traditionnelles et couvertures traditionnelles* ». Nous réalisons cette transformation en utilisant des expressions régulières. Au terme de cette phase, tout syntagme du texte correspondant à un concept des ressources est annoté de l'URI de celui-ci.

Nous exploitons les mêmes ressources, utilisées pour la représentation les données « métiers » dans le modèle d'EN *entreprise* (voir section 8.2), pour annoter les informations correspondantes. Ces ressources

sont construites à partir des hiérarchies NAF⁸⁶ et CFP⁸⁷ de l'INSEE décrivant les activités et produits d'entreprises, et de la hiérarchie ROME⁸⁸ de Pôle Emploi pour la description des métiers. Ces hiérarchies permettent de définir la taxonomie des ressources. Un processus d'enrichissement lexical par apprentissage supervisé est également mis sur pied pour ajouter des labels de vocabulaire à ces ressources. Nous détaillons ce processus dans la section 11.1.2 du chapitre 11.

Annotation des informations de contact - Les e-mails, les numéros de téléphone et de fax sont extraits en utilisant une approche basée sur des patrons. Ces patrons sont écrits en observant le même échantillon, des 150 sites web, utilisé pour l'extraction des adresses. Pour le cas des numéros de téléphone ou de fax, ceux-ci peuvent contenir un indicatif. De même, nous distinguons numéros de portables et numéros de téléphones fixes en utilisant des expressions régulières. Les numéros de fax sont considérés comme des numéros de téléphone fixes introduits par certains mots clés, comme « Télécopie », « Télécopieur », « Fax ». Les adresses, quant à elles, sont extraites selon l'approche détaillée dans la section 9.3.

Agrégation des propriétés d'EN *entreprise*

Dans le cadre du service *Cognisearch Business*, nous avons mis au point un processus de filtrage qui analyse le web et identifie le site web de chaque entreprise. Ainsi, le texte traité pour l'extraction d'information est constitué à partir des pages du site web d'une seule entreprise. De ce fait, les informations extraites sont relatives à une instance d'EN *entreprise*. Le processus d'agrégation de propriétés consiste donc à parcourir le texte et à associer les annotations détectées aux propriétés correspondantes dans notre modèle d'EN *entreprise*.

9.4.3 Expérimentation de l'extraction d'EN *entreprise*

La mise en œuvre de la chaîne décrite ci-dessus est expérimentée sur un corpus de textes de sites web d'entreprises. Nous avons vu que les informations d'une entreprise sont extraites à partir de son site web. Par conséquent, notre processus extrait autant d'EN *entreprise* que de sites web analysés. Ainsi, nous évaluerons les performances de notre système pour l'extraction des propriétés de l'EN *entreprise* à partir du texte de son site web. Cette évaluation portera notamment sur les propriétés adresse, e-mail, numéro de téléphone (fixe et portable) et activité. Les produits et métiers ne sont pas évalués parce que les ressources correspondantes sont en cours d'enrichissement.

9.4.3.1 Protocole d'évaluation de l'annotation des propriétés d'EN *entreprise*

Le processus d'évaluation est similaire à celui mis en œuvre pour les adresses. Il s'inspire également de la démarche TREC [179]. Dans notre cas, les « catégories » correspondent aux propriétés de l'EN *entreprise* que nous évaluons. Les « qrels » correspondent aux propriétés détectées par l'expert dans le texte du site web. La précision, le rappel et F₁-mesure sont évalués en comparant les « qrels » aux annotations posées par notre système. Le corpus d'évaluation est constitué de 100 pages d'accueil de sites web d'entreprises.

86. <https://www.insee.fr/fr/information/2406147>

87. <https://www.insee.fr/fr/metadonnees/cpfr21/section/A>

88. https://fr.wikipedia.org/wiki/Répertoire_opérationnel_des_métiers_et_des_emplois

9.4.3.2 Analyse des résultats de l’annotation des propriétés d’EN *entreprise*

Les résultats de l’expérimentation sont répertoriés dans le tableau 9.4. Pour chaque propriété, nous avons calculé les métriques obtenues sur le corpus d’évaluation.

Propriétés	Précision en %	Rappel en %	F ₁ -mesure en %
Adresse	89,09	83,52	86,21
E-mail	97,32	93,32	95,20
Numéro de téléphone	97,81	94,36	96,06
Activité	80,74	70,96	75,53

TABLE 9.4 – Évaluation de l’annotation des propriétés d’EN *entreprise*

Concernant l’annotation des adresses, notre système permet d’obtenir sur ce corpus une précision proche de 90%. Les annotations incorrectes sont dues en majorité à des adresses partiellement annotées. Des propriétés d’adresses (certains compléments d’adresse notamment), pourtant présentes dans le texte, ne sont pas détectées. Plus de 83% d’adresses détectées par l’expert sont correctement identifiées par notre système. Le silence observé est dû d’une part aux quelques formes d’adresses non couvertes et d’autre part à des erreurs d’analyse morpho-syntaxique des textes.

Pour les e-mails et les numéros de téléphone, nos annotateurs sont très performants (F₁-mesure très élevées). Très peu d’annotations sont incorrectes ce qui explique la très bonne précision. De même, nous observons que très peu d’e-mails ou de numéros de téléphone ne sont pas détectés par nos annotateurs. Ces bons résultats s’expliquent par le fait que ces propriétés suivent des formes très régulières, d’où l’efficacité de patrons conçus.

Plus de 80% des activités détectées par notre système sont correctes. Ce qui signifie que la projection des ressources sur le texte n’engendre pas beaucoup d’ambiguïté comme c’est le cas dans d’autres systèmes d’annotation basées sur des ressources ([158], [187]). De même, plus de 70% des activités d’entreprises renseignées sur leurs sites web sont détectées par notre système. Les activités non détectées correspondent généralement à des syntagmes du texte qui ne sont pas formulés de la même façon que dans la ressource. Dans un texte, nous pouvons par exemple avoir le syntagme « *charpente en bois traditionnel* » alors que dans la ressource, nous avons plutôt « *charpente traditionnelle en bois* ». Pour cet exemple, le texte ne sera pas annoté. Le processus d’enrichissement des ressources est nécessaire et se poursuit comme expliqué en section 11.1.2.

9.5 Extraction d’EN événement

Le module de prétraitements pour le service *Cognisearch Event* traite les pages d’un ensemble de plateformes du web dédiées aux événements. Ces pages sont réparties en deux catégories : (i) les pages « détail » qui décrivent un seul événement ; (ii) et les pages « autre » qui correspondent à des bref listings de séries d’événements, des pages de paiement ou toute page ne décrivant pas en détail un événement. Ce prétraitement est décrit en détail au chapitre 11, section 11.2.1. Seul le texte des pages « détail » est analysé par la suite pour l’extraction des EN *événement*. Le processus d’extraction exploite notre chaîne générique pour traiter le texte des pages. Nous détaillons dans cette section son instanciation pour traiter

le texte des pages « détail » ainsi que son expérimentation et les résultats obtenus.

9.5.1 Contexte d'extraction d'EN *événement*

Nous rappelons que le texte de chaque page « détail » ne décrit qu'un seul événement. De ce fait, certaines propriétés d'événements sont bruitées. Il s'agit notamment du titre ou des dates des représentations comme l'illustre l'exemple de la figure 9.1. En effet, la description de l'événement traité par la page fait référence au titre d'un autre événement du même artiste, ainsi qu'à la date de mise en vente des billets. De plus, certaines propriétés comme les lieux (adresse) de représentations peuvent être sur-composées ou exprimées suivant des formes non régulières (titre, intervenants, etc.). Nous instancions donc dans ce contexte notre chaîne générique pour identifier ces propriétés.

9.5.2 Mise en œuvre de la chaîne générique pour l'extraction d'EN *événement*

Des propriétés d'EN *événement* pouvant être bruitées dans les textes de notre corpus, il est donc nécessaire de localiser les blocs de texte dans lesquels elles sont situées avant d'invoquer les annotateurs. Ainsi, pour l'extraction des EN *événement*, nous instancions notre chaîne générique de la figure 9.3 avec tous les modules.

Détection des blocs contenant des propriétés

Infoconcert.com

YOUSSOU N'DOUR en concert : place de concert, billet, ticket et liste des concerts

Genre : World Music

Une des plus belles voix d'Afrique, doublée d'un artiste qui a su briser les barrières culturelles et économiques du show business international. Lire la présentation Dernière News : EVENEMENT / Le Bataclan annonce un début de programmation pour sa réouverture Acheter Prochains concerts Indiquez votre département Dernière mise en vente : BORGERHOUT le 23/11 Meilleure vente : PARIS le 15/11

Mardi 15 Novembre 2016 20h30	YOUSSOU N'DOUR À Paris (75) Philharmonie De Paris - Cite De La Musique de 33 à 55€	RÉSERVEZ VITE ! Ajout calendrier + d'infos sur le concert	bloc représentation 1
Vendredi 18 Novembre 2016 20h00	YOUSSOU N'DOUR À Paris (75) Bataclan de 37 à 50€	RÉSERVEZ VITE ! Ajout calendrier + d'infos sur le concert	bloc représentation 2
Samedi 19 Novembre 2016 20h00	YOUSSOU N'DOUR & ISMAEL LO À Paris (75) Bataclan de 37 à 50€	RÉSERVEZ VITE ! Ajout calendrier + d'infos sur le concert	bloc représentation 3
Mercredi 23 Novembre 2016 20h30	YOUSSOU N'DOUR À Borgerhout (Belgique) De Roma 36 €	RÉSERVEZ VITE ! Ajout calendrier + d'infos sur le concert	bloc représentation 4

Afficher les concerts de Youssou N'dour sur une carte Revendre et acheter d'occasion des places de concert pour Youssou N'dour Ajoutez un avis sur Youssou N'dour Ils peuvent aussi vous intéresser IRISH CELTIC 58 le 9 Novembre à Paris, CALYPSO ROSE 8 le 12 Octobre à Nancy ou SAAD LAMJARRED le 29 Octobre au Palais des Congrès de Paris INFOCONCERT SUR FACEBOOK Infoconcert.com ACCÈS RAPIDE AUX CONCERTS concert Youssou N'dour à Paris, concert Youssou N'dour à Paris, concert Youssou N'dour à Borgerhout.

FIGURE 9.11 – Détection de blocs dans le texte d'un événement contenant plusieurs représentations

En fonction de la propriété bruitée, nous utilisons soit des patrons, soit de l'apprentissage pour identifier le bloc de texte la contenant. Pour la détection du titre de l'événement, par exemple, nous formulons l'hypothèse suivante : « le titre d'un événement est contenu dans le texte correspondant à l'entête (*bloc structurel délimités par les balises <head>...</head>*) de la page ». En effet, nous avons constaté que pour des raisons de référencement naturel, le titre de l'événement décrit dans une page « détail » est très souvent contenu dans la balise <title> ou dans les méta-données de la page. Le « *bloc titre* » de la figure 9.11 correspond au texte d'entête d'une page « détail » du site infoconcert.com.

De même, lorsqu'un événement comporte plusieurs représentations, les informations de chacune d'elles sont regroupées au sein d'un même bloc de texte. Ces blocs contiennent généralement la date, le lieu, l'heure et parfois les intervenants ou les prix des billets. L'événement, décrit à la figure 9.11, est une tournée de l'artiste sénégalais « Youssou N'dour » en Europe. La première représentation (*bloc représentation 1*) de cette tournée à lieu le « 15 Novembre 2016 » au « Philharmonie de Paris » à partir de « 20 h 30 », le prix des billets variant entre 33 et 55 euros. Notre objectif dans ce module est de mettre en évidence de tels blocs dans le texte. Étant donné que les formes d'expression des représentations varient en fonction des plateformes et de la taille du texte analysé, nous mettons en œuvre une approche par apprentissage supervisé pour les détecter. Nous utilisons également l'apprentissage pour identifier les blocs correspondant aux catégories d'événements.

Annotation des propriétés d'EN événement

YOUSSOU N'DOUR en concert : place de concert, billet, ticket et liste des concerts

Aller au contenu principal YOUSSOU N'DOUR 0 0 Déposer un avis Genre : World Music Une des plus belles voix d'Afrique, doublée d'un artiste qui a su briser les barrières culturelles et économiques du show business international. Lire la présentation Dernière News : EVENEMENT / Le Bataclan annonce un début de programmation pour sa réouverture Acheter Prochains concerts. Indiquez votre département Dernière mise en vente : BORGERHOUT le 23/11 Meilleure vente : PARIS le 15/11

Mardi 15 Novembre 2016 20h30: YOUSSOU N'DOUR À Paris (75) Philharmonie De Paris - Cité De La Musique de 33 à 55€ RÉSERVEZ VITE ! Ajout calendrier + d'infos sur le concert

Vendredi 18 Novembre 2016 20h00: YOUSSOU N'DOUR À Paris (75) Bataclan de 37 à 50€ RÉSERVEZ VITE ! Ajout calendrier + d'infos sur le concert

Samedi 19 Novembre 2016 20h00: YOUSSOU N'DOUR & ISMAEL LO À Paris (75) Bataclan de 37 à 50€ RÉSERVEZ VITE ! Ajout calendrier + d'infos sur le concert

Mercredi 23 Novembre 2016 20h30: YOUSSOU N'DOUR À Borgerhout (Belgique) De Roma 36 € RÉSERVEZ VITE ! Ajout calendrier + d'infos sur le concert

Afficher les concerts de Youssou N'dour sur une carte Revendre et acheter d'occasion des places de concert pour Youssou N'dour Ajoutez un avis sur Youssou N'dour Ils peuvent aussi vous intéresser IRISH CELTIC 58 le 9 Novembre à Paris, CALYPSO ROSE 8 le 12 Octobre à Nancy ou SAAD LAMJARRED le 29 Octobre au Palais des Congrès de Paris INFOCONCERT SUR FACEBOOK Infoconcert.com ACCÈS RAPIDE AUX CONCERTS concert Youssou N'dour à Paris, concert Youssou N'dour à Paris, concert Youssou N'dour à Borgerhout.

FIGURE 9.12 – Annotation des propriétés d'événements

Après avoir mis en évidence les blocs de texte correspondants aux propriétés de l'événement, nous exploitons différentes approches d'extraction pour les annoter. La figure 9.12 illustre les annotations

obtenues en sortie de ce module sur un extrait de texte de la page « détail » correspondant à la tournée de « Youssou N'dour ». Ces annotations correspondent aux rectangles en pointillés bleus de la figure.

Dans ce paragraphe, nous détaillons l'annotation des propriétés : titre, catégories, lieux, dates et intervenants.

Annotation du titre - Pour l'identification du titre de l'événement dans le bloc de texte correspondant à l'entête des pages, nous optons pour une approche basée sur l'apprentissage supervisé. Le choix de l'apprentissage se justifie par le fait que le titre de l'événement ne suit pas de formes régulières ou n'est pas exprimé selon un vocabulaire contrôlé. Il peut s'agir, comme nous l'avons déjà mentionné, d'un seul mot (« Soprano »), d'un groupe de mots (« Jeff Panacloc contre-attaque ») ou même d'une phrase (« Romanès cirque tsigane : les Nomades tracent les chemins du ciel! »). De ce fait, nous ne pouvons pas utiliser les patrons et encore moins des ressources pour l'identifier. Comme précisé en section 5.2.2 de l'état de l'art, l'apprentissage est adapté pour identifier la propriété titre dans notre contexte.

Annotation des catégories - Elle exploite une approche basée sur des ressources externes. Ce choix se justifie par le fait que le vocabulaire utilisé pour exprimer les catégories d'événements socio-culturels est contrôlé. Nous avons analysé un ensemble de plateformes en ligne traitant de ce type d'événements pour constituer cette ressource, notamment les sites de billetterie, les sites des offices de tourisme, les sites de communes et les annuaires d'événements. Nous représentons la ressource des catégories sous forme d'un arbre de concepts ; chaque concept étant caractérisé par un ensemble de labels. L'arbre obtenu a 4 niveaux hiérarchiques et contient 57 concepts pour 610 labels. La ressource est projetée sur les blocs de texte pour annoter les mentions de catégories d'événements.

Annotation des lieux de représentations - L'approche mise en œuvre pour l'identification des lieux est de type hybride, elle se décline en trois phases.

- Le texte est d'abord analysé avec l'approche d'extraction d'adresse présentée à la section 9.3. L'objectif de cette première phase est d'extraire le lieu sous sa forme la plus complète (*Salle Eurythmie, Rue Salvador Allende 82000 Montauban*).
- Si la phase précédente n'a pas permis de détecter une adresse, nous exploitons une ressource pour annoter le lieu de la représentation. En effet, à partir des données de DBPedia et de Google Maps, nous avons constitué une ressource de lieux susceptibles d'accueillir un événement. Il s'agit notamment des salles de spectacles, des arènes, des stades ou complexes sportifs, des salles de conférence. La ressource construite est représentée sous forme d'un graphe de concepts stocké au format OWL. Le choix de ces deux ressources se justifie par le fait qu'elles sont riches et mises à jour régulièrement.
- Si aucune des deux phases précédentes n'a détecté de lieu, nous utilisons le dictionnaire des noms de ville pour annoter le bloc de texte.

La combinaison de ces trois phases pour annoter les lieux se justifie par le fait qu'ils ne sont pas exprimés de la même façon dans les différents sites web de notre corpus. Alors que dans certains sites c'est l'adresse qui renseigne le lieu, dans d'autres c'est le nom de la salle ou simplement le nom de la commune. Pour l'exemple des représentations de la figure 9.12, la phase 1 n'annotera aucune adresse, c'est la phase 2 qui permet d'identifier les différents noms de salle.

Annotation des dates de représentations - L'approche mise en œuvre ici est basée sur les patrons. Ce choix s'explique par le fait que les dates dans les textes de notre corpus suivent des formes très régulières et limitées. Généralement, une date est exprimée par un jour, un mois et une année. Les mois sont parfois écrits en toutes lettres (« juin ») ou simplement par un numéro (« 06 » ou « 6 »). Nous avons constitué un dictionnaire pour identifier les mois écrits en lettres, ainsi que leurs différentes abréviations.

Annotation des intervenants - Nous mettons en œuvre une approche hybride combinant ressources externes et apprentissage supervisé. Le processus d'annotation se décline en deux phases.

- Une ressource de type connaissance, dédiée aux intervenants d'événements, est projetée sur le texte. Nous avons construit cette ressource à partir de DBpedia et Yago⁸⁹. En utilisant des requêtes SPARQL, nous avons extrait les catégories suivantes : artistes, organisations, personnalités (politique, scientifique, etc.), groupes (musicaux, troupes théâtrales, etc.), équipes (sportives, professionnelles, etc.). Nous avons représenté cette ressource sous forme d'un graphe de concepts stocké au format OWL.
- Un modèle d'apprentissage, construit de façon supervisé, est également appliqué au texte pour détecter d'autres noms d'intervenants. Nous avons mis en œuvre cette deuxième phase pour identifier les noms d'intervenants éventuels qui ne seraient pas référencés dans la ressource. En effet, certains artistes ou certains groupes « peu connus » ne sont que rarement référencés dans les ressources du web. De ce fait, cette deuxième phase permet d'atténuer le silence.

Agrégation des propriétés d'EN *événement*

Dans notre contexte, le texte d'une page « détail » traite a priori d'une seule instance d'EN *événement*. De ce fait, comme pour les entreprises, notre processus d'agrégation consiste à parcourir le texte pour repérer des annotations de propriétés et à les associer aux champs correspondants du modèle.

Concernant les représentations, nous exploitons les marquages de blocs correspondants pour agréger leurs propriétés. Ainsi, le texte de chaque bloc « *représentation* » est analysé pour associer les annotations contenues aux propriétés correspondantes.

La mise en œuvre de l'apprentissage pour la détection des blocs et l'annotation des propriétés est décrit en détail à la section 11.2.2 du chapitre 11. Nous y présentons l'algorithme, ses paramètres, ainsi que le corpus d'entraînement et ses caractéristiques.

9.5.3 Expérimentation de l'extraction d'EN *événement*

Nous expérimentons la mise en œuvre de la chaîne générique décrite à la section précédente pour annoter des propriétés d'EN *événement* sur un corpus de pages issues d'un ensemble de sites de billetterie. Dans notre contexte, le prétraitement des pages du corpus, permet de trier les pages « détail » dont le texte est analysé pour extraire les informations. Par conséquent, nous extrayons autant d'EN *événement* qu'il y a de pages « détail » analysées. C'est la raison pour laquelle nous évaluons la capacité de notre système à annoter correctement des propriétés d'EN *événement* dans le texte.

Notre corpus d'évaluation est constitué de 150 pages « détail » tirées de 12 sites de billetterie.

89. <http://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago/#c10444>

9.5.3.1 Protocole d'évaluation de l'annotation des propriétés d'EN *événement*

Le module d'annotation de propriétés, puisque nous sommes dans un contexte bruité, dépend fortement des résultats du module de détection de blocs. De ce fait, nous commencerons par évaluer ce module avant d'évaluer l'annotation des propriétés.

Pour cette deuxième évaluation, nous distinguerons deux scénarios, l'objectif visé est d'évaluer l'apport du module de détection de blocs pour l'extraction d'événements :

- pour le premier, nous évaluons les annotateurs sans le module de détection de blocs ;
- pour le deuxième, l'évaluation se fait avec ce module.

Nous nous inspirons de la démarche TREC [179] pour cette évaluation. La précision, le rappel et la F_1 -mesure sont calculés pour chaque type de blocs et pour chaque propriété.

9.5.3.2 Analyse de résultats de l'annotation de blocs et de propriétés

Détection de blocs - Nous avons expérimenté notre module de détection de blocs pour la localisation des blocs *catégorie* et *représentation*. Les résultats obtenus sont présentés dans le tableau 9.5.

Blocs ciblés	Précision en %	Rappel en %	F_1 -mesure en %
<i>Représentation</i>	94,58	83,16	88,50
<i>Catégorie</i>	93,25	82,17	87,36

TABLE 9.5 – Évaluation du module de détection de blocs

Ces résultats montrent que notre module de détection de blocs est très précis. Ainsi, lorsqu'il détecte un bloc de texte comme contenant une propriété bruitée, ce marquage est correct dans plus de 93% des cas. Ce qui est un très bon résultat puisque des marquages de blocs erronés engendreraient automatiquement des erreurs dans l'annotation des propriétés correspondantes.

De même, peu de blocs contenant des propriétés bruitées sont manqués par notre module. Pour le cas des représentations, les blocs qui ne sont pas détectés correspondent aux représentations qui se tiennent au même endroit, mais à des dates et heures différentes. L'extrait de description d'événement suivant, met en évidence ce type de représentations (en italique) « (...) au Bikini les jours suivants : le *16 juin*, à partir de 20 h (...), le *20 juin*, à partir de 18 h (...) ». Ce texte contient plusieurs représentations pour les différents couples de dates et d'heures alors que notre système les intègre dans un seul bloc. Pour les catégories, la variation du vocabulaire utilisé pour les exprimer est la principale cause de la non détection des blocs correspondants.

Concernant les blocs *titre*, notre module les détecte correctement dans toutes les pages du corpus d'évaluation. Ceci s'explique par le fait qu'il est simple de détecter l'entête d'une page HTML et par conséquent le texte associé.

Annotation de propriétés sans détection préalable de blocs - Nous avons expérimenté l'identification des propriétés d'événements en invoquant directement les annotateurs correspondants sur tout le texte. Les résultats obtenus sont présentés dans le tableau 9.6.

Propriétés	Précision en %	Rappel en %	F ₁ -mesure en %
<i>Titre</i>	75,26	48,67	59,11
<i>Catégorie</i>	42,10	92,05	57,77
<i>Lieu</i>	51,02	95,28	66,46
<i>Date</i>	52,82	94,62	67,79
<i>Intervenant</i>	37,15	90,24	52,63

TABLE 9.6 – Évaluation du module d'annotation de propriétés sans détection préalable de blocs

Nous remarquons que pour les propriétés suivant des formes régulières (lieux et dates de représentations) et pour celles dont le vocabulaire peut être répertorié dans des ressources (catégorie et intervenant), les rappels obtenus sont très bons. Cependant, les précisions sont médiocres, ceci parce que toutes les annotations posées ne sont pas forcément pertinentes en raison du bruit. Dans le cas des dates, par exemple, le processus expérimenté ici annote toutes les dates comme des propriétés d'événement, sans tenir compte du fait que le texte puisse faire référence, par exemple, à la date de mise en vente des billets (voir figure 9.1). Dans le cas des catégories ou des intervenants, la projection de la ressource annote tous les syntagmes qui correspondent à des labels des ressources : d'où le taux élevé d'annotations incorrectes.

Concernant les titres d'événements, la précision est moyenne et le rappel est faible. L'annotateur rate quasiment un titre sur deux. En effet, le grand nombre de formes de titres et le fait d'analyser tout le texte impacte les performances du modèle d'apprentissage. Par conséquent, certains titres d'événements ne sont pas annotés ou bien certaines annotations correspondent bien à des titres d'événements, mais n'ont rien à voir avec celui décrit dans la page.

Annotation de propriétés avec détection préalable de blocs - Nous avons expérimenté le processus d'annotation de propriétés d'événements en exploitant les blocs détectés par le premier module de notre chaîne générique. Le tableau 9.7 présente les résultats obtenus pour cette expérimentation (P' , R' et F'_1). Dans ce même tableau, nous reprenons les résultats du scénario d'évaluation précédent (P , R et F_1) pour mesurer les gains obtenus avec le module de détection de blocs (Δ_P , Δ_R et Δ_{F_1}).

Propriétés	Scénario 1			Scénario 2			Gains		
	P	R	F_1	P'	R'	F'_1	Δ_P	Δ_R	Δ_{F_1}
<i>Titre</i>	75,26	48,67	59,11	92,57	87,63	90,03	+ 17,31	+ 38,96	+ 30,91
<i>Catégorie</i>	42,10	92,05	57,77	92,45	76,15	83,51	+ 50,35	- 15,90	+ 25,73
<i>Lieu</i>	51,02	95,28	66,46	94,25	81,16	87,22	+ 43,23	- 14,12	+ 20,76
<i>Date</i>	52,82	94,62	67,79	95,82	82,03	88,39	+ 43,00	- 12,59	+ 20,59
<i>Intervenant</i>	37,15	90,24	52,63	84,16	79,24	81,63	+ 47,01	- 11,00	+ 28,99

TABLE 9.7 – Évaluation du module d'annotation de propriétés avec détection préalable de blocs

Le premier constat qui se dégage de l'observation de ces résultats est la nette amélioration de toutes les F₁-mesures avec l'instanciation du module de détection de blocs. Toutefois, cette amélioration se fait en impactant le rappel pour certaines propriétés, notamment la catégorie, le lieu, la date et les intervenants.

En effet, si le bloc contenant une propriété bruitée n'a pas été détecté, alors la propriété associée ne sera pas annotée, d'où l'impact sur le rappel.

Concernant l'annotation du titre de l'événement, l'instanciation du module de marquage de blocs permet d'augmenter à la fois la précision et le rappel. En effet, traiter uniquement le texte de l'entête des pages réduit considérablement le risque d'annoter un autre titre à tort : ceci explique la hausse de la précision. Par ailleurs, le fait de n'analyser qu'un fragment de texte permet de tirer profit des performances de l'apprentissage pour l'annotation d'information dans des textes courts ([149], [109], [104]). De ce fait, le modèle d'apprentissage obtenu est plus performant et donc annoté plus de titres, d'où le gain en rappel. Toutefois, certains titres d'événements ne sont parfois pas détectés par notre annotateur. Il s'agit généralement des titres très courts (« P. Kass »).

En ce qui concerne la propriété catégorie, nous constatons que la précision est de 92% pour le scénario d'évaluation qui intègre le module de détection de blocs. Ce résultat est particulièrement intéressant dans la mesure où il est meilleur que celui obtenu pour des systèmes d'annotation d'EN s'appuyant sur des ressources externes (65% en moyenne pour les systèmes ciblant des EN dans le domaine médical [158], [187]). De plus, le module de détection de blocs améliore cette précision de 50% par rapport au scénario 1. Cependant, l'annotateur de catégories ne détecte parfois pas de labels dans les blocs sélectionnés. Ceci s'explique principalement par le fait que les syntagmes correspondant ne sont pas répertoriés dans la ressource.

Pour les lieux, la précision obtenue est d'environ 95%, bien au delà des 83% obtenus en moyenne par Jiang et al. [86], qui ont expérimentés plusieurs SREN libres (Stanford NER⁹⁰ et Spacy⁹¹ notamment) sur différents corpus de textes. L'annotation en plusieurs phases limite ainsi la proportion de lieux non identifiés. Par conséquent, presque tous les lieux dans les blocs représentations sont annotés par notre système.

Pour les dates, le module de détection de blocs permet d'isoler correctement les dates pertinentes. Ce qui permet de limiter considérablement la proportion de dates erronées, d'où l'amélioration de la précision (gain de 43%). Cette précision est du même ordre que celle obtenue par Li et al. [105] qui utilise le système HeidelTime⁹² pour annoter des informations temporelles dans les textes chinois.

Concernant les intervenants, la précision de notre système est de 84%. Ce qui est un bon résultat, comparé aux 75% obtenus par Jiang et al. [86] pour l'annotation des noms de personnes dans différents corpus de textes en utilisant les SREN Stanford NER⁹³ et Spacy⁹⁴. Les annotations erronées correspondent généralement à des cas ambigus. Considérons que dans la ressource, par exemple, nous avons un intervenant dont un label est « *Jean Marc* » (la peluche de Jeff Panaoloc), mais que le texte porte sur la description d'une conférence animée par un certain « *Jean Marc Verdier* » qui n'est pas référencé dans la ressource. Notre système annotera le syntagme « *Jean Marc* » (la peluche) qui n'a absolument rien à voir avec la conférence.

Nous souhaitons toutefois préciser qu'il nous faut poursuivre ces expérimentations sur des corpus plus importants, afin de confirmer ces premiers résultats encourageants et nos comparaisons aux divers systèmes d'annotation existants.

90. <https://nlp.stanford.edu/ner/>

91. <https://spacy.io/>

92. <https://code.google.com/archive/p/heideltime/>

93. <https://nlp.stanford.edu/ner/>

94. <https://spacy.io/>

9.6 Bilan

Dans ce chapitre, nous présentons notre contribution portant sur l’extraction d’EN complexes sur le web. Il s’agit d’une chaîne générique dédiée à l’analyse de textes issus du web pour extraire des EN complexes indépendamment de leur type. Plusieurs verrous sont pris en compte pour l’identification des propriétés. Ils correspondent notamment à l’extraction d’information dans un contexte bruité, à l’annotation d’entités dont les formes d’expression ne sont pas régulières, au traitement d’EN dont les syntagmes correspondant dans le texte sont sur-composées.

L’originalité de cette chaîne vient du fait qu’elle permet de traiter suivant le même processus, différents types d’EN complexes. Elle est constituée de trois principaux modules de traitements : (i) le premier a pour but de détecter les blocs de texte délimitant des propriétés ; (ii) le deuxième analyse ces blocs pour annoter les propriétés d’EN qu’ils contiennent ; (iii) le dernier module regroupe les annotations de propriétés pour construire des instances d’EN. Ce module tient compte du fait que le texte analysé puisse contenir plusieurs instances distinctes de l’EN ciblée. L’introduction du module de détection de blocs est une contribution intéressante qui constitue une piste d’amélioration des systèmes d’extraction d’EN. Nousinstancions notre chaîne, pour extraire des EN *adresse*, *entreprise* et *événement*.

Dans le cas particulier des adresses, nous avons proposé deux annotateurs basés sur des patrons pour identifier le complément d’adresse et le nom de la voie dans un contexte où les propriétés de l’adresse peuvent être permutées et sur-composées. Notre approche est indépendante de ressources propriétaires et exploite des dictionnaires pour identifier les syntagmes nécessaires pour le déclenchement des patrons. les résultats de nos expérimentations sont encourageants. Ils sont même meilleurs que ceux obtenus par d’autres systèmes ([162]) qui s’intéressent à l’extraction d’adresses dans un contexte proche du nôtre.

Dans le cas de l’extraction d’EN *événement* qui se fait dans un contexte bruité, nous avons expérimenté deux scénarios : le premier invoque directement les annotateurs de propriétés et le deuxième instancie d’abord le module de détection de blocs avant d’invoquer les annotateurs. Les résultats de l’expérimentation de ces deux scénarios permettent de constater une nette amélioration des performances du système lorsque le module de détection de blocs est instancié.

Nos travaux portant sur l’extraction d’EN complexes dans un texte ont été validés par des publications et des communications :

- à la conférence AllData [59] en Février 2016 à Lisbonne ;
- à la conférence iSWAG [58] en Juin 2016 à Deauville ;
- à l’atelier GAST [56] en Janvier 2016 à Reims.

Chapitre 10

Calcul de similarité entre entités nommées complexes

Les modules *indexation* et *recherche d'information* de l'architecture *Cognisearch* s'appuient tous les deux sur le calcul de similarité pour : (i) intégrer sans doublon de nouvelles EN extraites du web ; (ii) et pour appairer les besoins d'information exprimés par les utilisateurs aux EN indexées. De ce fait, nous proposons des approches pour évaluer de façon générale le score de similarité entre EN complexes. Nous expérimentons ces approches pour le calcul de similarité entre deux types d'EN complexes.

Ce chapitre s'organise en trois grandes parties : (i) une première rappelle notre contexte de travail, tout en positionnant nos travaux par rapport aux approches de l'état de l'art ; (ii) une deuxième partie présente nos propositions pour le calcul de similarité entre EN complexes ; (iii) et une dernière partie, détaille la mise en œuvre et l'expérimentation de ces propositions.

10.1 Rappel du contexte et positionnement du travail

Les informations d'une même EN complexe peuvent être référencées sur plusieurs sources distinctes du web. C'est le cas par exemple du festival « Main Square 2017 » illustré par la figure 10.1. Nous l'avons identifié sur deux plateformes distinctes du web : infoconcert.com et ticketmaster.com. Les informations contenues sur ces deux plateformes ne sont pas exprimées de la même façon (titre, lieu, etc.). À partir de chacune des plateformes, le module d'*extraction d'information* construira deux EN distinctes et pourtant il s'agit du même événement.

D'autre part, deux pages d'un même site peuvent décrire des représentations différentes d'un même événement. C'est le cas, par exemple, de la tournée de « Florent Pagny », dont deux représentations distinctes sur le site ticketmaster.com sont illustrées par la figure 10.2. Notre module d'*extraction d'information* traite les deux pages séparément et donc en extrait deux EN. Au moment de l'indexation, il faut fusionner ces deux EN pour construire une seule EN *événement*, avec deux représentations correspondant aux deux concerts présentés à Lille et à Toulouse.

Par ailleurs, le module de *recherche d'information* que nous avons conçu représente le besoin d'information comme une EN *événement* dans laquelle certaines propriétés peuvent être non renseignées. L'opération d'appariement compare cette EN-requête avec celles indexées pour déterminer les plus per-

Chapitre 10. Calcul de similarité entre entités nommées complexes

The figure shows two screenshots of a website event page for 'MAIN SQUARE FESTIVAL 2017'. The top screenshot is from 'INFOCONCERT.COM' and the bottom is from 'ticketmaster.com'. Both pages display the event title, dates (30 June to 2 July 2017), location (Arras), and a list of artists including Radiohead, Major Lazer, and System of a Down. The ticketmaster page also shows a '1 - Réservation' button and a 'billetcollector' logo.

FIGURE 10.1 – Exemple d'événement référencé sur deux sites différents

The figure shows two screenshots of the same event page for 'FLORENT PAGNY'. The top screenshot shows the event on '13 oct. 2017' at 'ZENITH DE TOULOUSE'. The bottom screenshot shows the event on '22 sept. 2017' at 'ZENITH ARENA LILLE'. Both pages display the artist's name, event type (concert), and a 'billetcollector' logo. The content is identical, showing the artist's 55th anniversary tour.

FIGURE 10.2 – Exemple d'un événement avec deux représentations décrites sur deux pages distinctes d'un même site web

tinentes.

En somme, le calcul de similarité entre EN complexes est un verrou principal pour nos modules d'indexation et de recherche d'information. Il pose de nombreuses difficultés à deux niveaux pour notre contexte : au niveau de la comparaison des propriétés des EN et au niveau de l'agrégation des similarités entre valeurs de propriétés au même rang pour en déduire la similarité globale de deux EN. Pour la similarité entre propriétés, nous constatons, par exemple, que celles-ci ne sont pas toujours exprimées de la même façon ou avec le même niveau de précision dans les pages traitées. En reprenant l'exemple de la figure 10.1, nous observons que sur le site infoconcert.com, le lieu de la représentation est donné par le nom de ville (« Arras »), alors que sur ticketmaster.com nous avons une adresse complète (« La Citadelle - Quartier de Turenne, Bvd du Général de Gaule, 62000 Arras - France »). La ville est représentée dans notre modèle par un polygone alors que l'adresse est donnée par un point. La difficulté ici est d'évaluer cette similarité en tenant compte du fait qu'un point peut être localisé dans la périphérie d'une ville.

Au niveau de l'agrégation des similarités entre propriétés (similarités intermédiaires), une difficulté majeure dans notre contexte découle du fait que certaines propriétés peuvent ne pas être renseignées dans les instances d'EN comparées. L'exemple de la figure 10.1 illustre un tel cas, sur ticketmaster, la catégorie de l'événement est explicitement spécifiée, ce qui n'est pas le cas sur infoconcert. Tout l'enjeu est de réussir à établir la similarité entre les EN extraites, en tenant compte de ce manque potentiel d'information.

Face à ces difficultés, nous formulons quelques propositions. En ce qui concerne le calcul de similarité entre propriétés de même rang, et plus particulièrement, le calcul de la similarité entre lieux, l'un représenté par un point et l'autre par un polygone, la non-similarité est généralement établie si le point n'est pas inclus dans le polygone [152]. De ce fait, *l'université de Bordeaux-Montaigne* et la ville de Bordeaux seraient considérés comme non similaires, simplement parce que cette université est située dans la périphérie de la ville Bordeaux. Nous proposons, pour ce cas particulier, une fonction de calcul de similarité adaptée.

Concernant l'agrégation des différentes similarités intermédiaires, l'approche la plus utilisée est la combinaison linéaire (somme pondérée). Les poids pour les différentes propriétés sont habituellement définis de façon empirique ([165], [142], [185], [84], [12], etc.). Nous proposons également des approches basées sur la combinaison linéaire, dont les poids sont déterminés pour l'une par étalonnage et pour l'autre par apprentissage. En outre, l'utilisation de la combinaison linéaire pour agréger les similarités intermédiaires suppose que celles-ci soient linéairement indépendantes. Cependant, pour des EN complexes de type *entreprise*, par exemple, la similarité entre les services offerts peut être liée à la similarité entre leurs métiers. Afin de tenir compte du cas où cette hypothèse d'indépendance ne peut être garantie, nous proposons également des approches de calcul de similarité, entre EN complexes, valides qu'il y ait dépendance ou pas entre similarités intermédiaires.

10.2 Proposition de quatre approches de calcul de similarité entre EN complexes

Nous détaillons dans cette section nos quatre approches de calcul de similarité entre EN complexes. Trois d'entre elles sont dédiées à l'agrégation des similarités intermédiaires, et la dernière combine l'organisation des EN en clusters et leur représentation vectorielle pour le calcul de similarité.

Soient deux EN complexes o_1 et o_2 ayant n propriétés et définies par :

$$o_1 = \langle p_{11}, p_{12}, \dots, p_{1n} \rangle$$

$$o_2 = \langle p_{21}, p_{22}, \dots, p_{2n} \rangle$$

Chaque propriété peut être monovaluée ou multivaluée (ensemble de valeurs). De ce fait, elle peut correspondre à un type simple (texte, date, heure, lieu, concept d'ontologie), un ensemble d'éléments de type simple, une autre EN complexe ou un ensemble d'EN complexes. On note T_i le type de la propriété au rang i . Avant de présenter en détail chacune de nos approches de calcul de similarité entre EN complexes, nous commençons par présenter les mesures de similarité entre propriétés.

10.2.1 Calcul de similarité entre propriétés

Pour les propriétés au rang i de o_1 et o_2 , le calcul de leur similarité dépend de leur type. De façon générale, f_{T_i} la fonction qui permet de calculer le score de similarité entre p_{1i} et p_{2i} est définie par :

$$\begin{aligned} f_{T_i} : T_i \times T_i &\longrightarrow [0, 1] \\ (p_{1i}, p_{2i}) &\longmapsto s_i(p_{1i}, p_{2i}) \end{aligned}$$

10.2.1.1 Calcul de similarité entre propriétés de type simple

Nous nous intéressons au cas où p_{1i} et p_{2i} sont représentées par du texte court, par des concepts d'ontologies ou par des valeurs numériques. Ce dernier cas englobe les EN temporelles (dates et heures) et les lieux.

Calcul de $f_{\text{texte_court}}$ - Nous appelons **texte_court**, une chaîne de caractères constituée d'une quinzaine de mots au maximum. Pour évaluer le degré de similarité entre textes courts, nous ré-utilisons la métrique *softTFIDF* [32] (cf. chapitre 6 de l'état de l'art). Ce choix se justifie par le fait que *softTFIDF* soit une métrique hybride. De ce fait, le calcul de similarité, compare les mots, mais aussi les caractères qui les composent. Par conséquent, ce calcul est souple vis à vis des erreurs d'orthographe. De plus, contrairement aux autres métriques hybrides de calcul de similarité entre chaînes de caractères, *softTFIDF* ne tient pas compte de l'ordre des mots. Ainsi les deux chaînes « Main Square Festival » et « Festival Main Square » sont considérées identiques.

Calcul de $f_{\text{semantique}}$ - En ce qui concerne les propriétés représentées par les concepts d'ontologies, nous utilisons la mesure de Wu et Palmer [186] (cf. chapitre 6 de l'état de l'art) pour calculer leur degré de similarité. Ce choix se justifie par le fait que cette mesure permet de prendre en compte la taxonomie (hiérarchie) de l'ontologie [48].

Soit $\Omega = \langle N, A \rangle$ l'arbre correspondant à la taxonomie de l'ontologie utilisée pour représenter p_{1i} et p_{2i} . N est l'ensemble des concepts de l'ontologie (les nœuds de Ω) et A est l'ensemble des relations sémantiques entre concepts (les arêtes de Ω).

Soient c_1 et $c_2 \in N$, deux concepts de l'ontologie et $c \in N$ leur plus proche ancêtre commun.

On appelle *profondeur* la fonction qui évalue la profondeur d'un concept, c'est-à-dire le nombre d'arêtes entre ce concept et le nœud racine de l'arbre. La fonction $f_{\text{semantique}}$ est définie par :

$$f_{\text{semantique}}(c_1, c_2) = \frac{2 \cdot \text{profondeur}(c)}{\text{profondeur}(c_1) + \text{profondeur}(c_2)}$$

Calcul de f_{temps} - Dans ce cas, les deux EN temporelles t_1 et t_2 sont représentées par des points sur l'axe du temps. On désigne par θ l'amplitude (longueur de l'intervalle séparant deux points) à partir de laquelle deux EN temporelles sont considérées comme non similaires. La fonction f_{temps} est définie de la même façon que dans [12] (cf. chapitre 6 de l'état de l'art) et est donnée par la formule :

$$f_{\text{temps}}(t_1, t_2) = \begin{cases} 0 & \text{si } |t_2 - t_1| > \theta \\ 1 - \frac{|t_2 - t_1|}{\theta} & \text{sinon.} \end{cases}$$

NB : la valeur de θ est déterminée en fonction du corpus de données et du type d'entités temporelles traitées (dates ou heures).

Calcul de f_{lieu} - Les lieux sont représentés par deux géométries : les points (couples latitude-longitude) ou les polygones (liste de points). Le score de similarité $f_{\text{lieu}}(l_1, l_2)$ entre deux lieux l_1 et l_2 dépend de la géométrie représentant chacun d'eux. Trois cas sont possibles.

— **Cas 1 : l_1 et l_2 sont des points de l'espace**

On désigne par H la fonction de *Haversine* [31] (cf. chapitre 6 de l'état de l'art) permettant de calculer la distance entre deux points de la surface de la terre et γ la distance seuil à partir de laquelle deux points sont considérés comme totalement distincts.

Pour le calcul de la similarité entre l_1 et l_2 , nous réutilisons la fonction proposée dans [12]. Ainsi, f_{lieu} est définie par :

$$f_{\text{lieu}}(l_1, l_2) = \begin{cases} 0 & \text{si } H(l_1, l_2) > \gamma \\ 1 - \frac{H(l_1, l_2)}{\gamma} & \text{sinon.} \end{cases}$$

— **Cas 2 : l_1 et l_2 sont des polygones**

On note *surface* la fonction permettant de calculer la surface d'un polygone, Nous définissons f_{lieu} en fonction de la surface de recouvrement [182] (cf. chapitre 6 de l'état de l'art) entre les polygones l_1 et l_2 .

$$f_{\text{lieu}}(l_1, l_2) = \frac{\text{surf}(l_1 \cap l_2)}{\text{surf}(l_1 \cup l_2)}$$

En effet, nous avons choisi une fonction basée uniquement sur le recouvrement parce que les polygones dans notre contexte correspondent à des communes. De ce fait, elles ont des formes bien définies et sont disjointes à priori. Par conséquent, nous n'avons pas besoin de prendre en compte leurs surfaces respectives dans le calcul de la similarité.

— **Cas 3 : l_1 est un point et l_2 est un polygone**

Pour ce cas, nous souhaitons évaluer la similarité en intégrant le fait que, le point puisse être contenu dans la périphérie de la zone définie par le polygone. L'approche que nous proposons pour ce cas est une contribution qui complète celle proposée par Rueben et al. [152].

Nous désignons par P le polygone étendu de l_2 et c_P son centroïde. P est construit à partir de l_2 en appliquant une translation d'une distance d à chacun de ses sommets tout en gardant sa forme

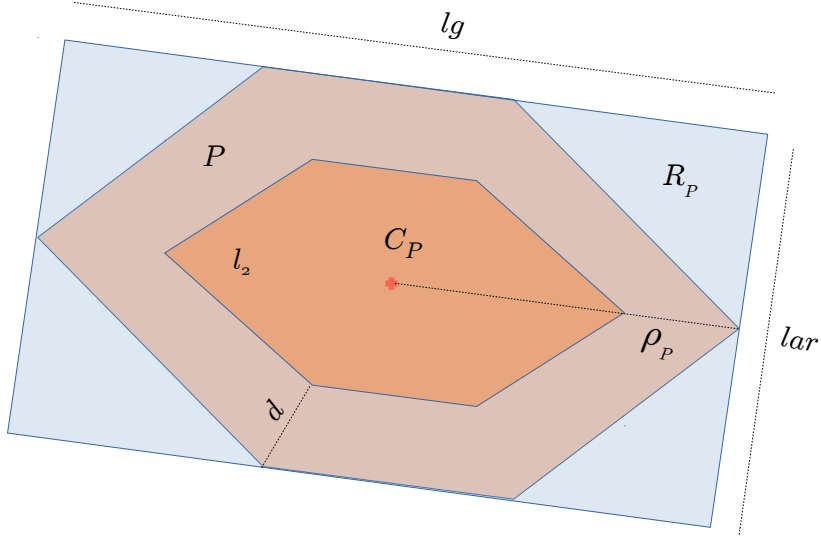


FIGURE 10.3 – Rectangle englobant un polygone

initiale (voir figure 10.3). On note ρ_P la distance maximale entre c_P et chacun des sommets de P .

$$\rho_P = \max(H(c_P, \text{point}), / \text{point} \in P)$$

La distance d permettant de définir l'extension du polygone est paramétrable.

On définit dans ce cas la fonction f_{lieu} par :

$$f_{lieu}(l_1, l_2) = \begin{cases} 0 & \text{si } l_1 \notin P \\ 1 - \nu \cdot \frac{H(l_1, c_P)}{\rho_P} & \text{sinon.} \end{cases}$$

ν , avec $0 < \nu < 1$ est un coefficient permettant de prendre en compte l'étirement (en longueur ou en largeur) du polygone P dans le calcul de similarité. Ce coefficient est calculé à partir du rectangle englobant R_P de P (voir figure 10.3). R_P est choisi de telle sorte que la surface résiduelle entre R_P et P soit minimale. On note lar et lg la largeur et la longueur respectivement de R_P . ν est donnée par la formule :

$$\nu = \left(\frac{lar}{lg}\right)^2$$

NB : Le calcul de f_{lieu} se fait exactement de la même façon dans le cas où l_2 est un point et l_1 un polygone.

10.2.1.2 Calcul de similarité entre propriétés de type ensemble

Dans ce cas, T_i correspond à un ensemble d'éléments. Il peut s'agir par exemple d'un ensemble de concepts d'ontologie, d'un ensemble de textes courts, d'un ensemble de lieux, d'un ensemble d'entités temporelles ou même d'un ensemble d'EN complexes. Nous ré-utilisons la fonction proposée par Halkidi

et al. [75] pour le calcul de la similarité entre ensemble d'éléments. Nous rappelons que cette fonction est dédiée au calcul de la similarité entre ensemble de concepts en exploitant la mesure de Wu et Palmer [186]. Pour notre problème, nous remplaçons la mesure de Wu et Palmer par une fonction adaptée au calcul de la similarité entre éléments d'ensembles.

Soient E_1 et E_2 deux ensembles, dont les n et m éléments respectivement sont du type T .

$$E_1 = \{u_1, u_2, \dots, u_n\}$$

$$E_2 = \{v_1, v_2, \dots, v_m\}$$

Le score de similarité entre ces deux ensembles est déterminé par la formule :

$$f_{ensemble}(E_1, E_2) = \frac{1}{2} \left(\frac{1}{n} \sum_{i=1}^n \max_{1 \leq j \leq m} f_T(u_i, v_j) + \frac{1}{m} \sum_{j=1}^m \max_{1 \leq i \leq n} f_T(v_j, u_i) \right)$$

Nous avons choisi cette fonction au détriment d'autres fonctions de similarité entre ensembles comme celle de Jaccard. En effet, la fonction de Jaccard compare deux ensembles, en évaluant seulement le rapport entre les cardinalités de leur intersection et de leur réunion. De ce fait, s'il y a une grande différence entre les cardinalités des deux ensembles, la similarité sera d'office faible même si l'un des ensembles est inclus dans l'autre. La fonction proposée par Halkidi et al., basée sur le produit cartésien des éléments des deux ensembles, permet de tenir compte de la différence de cardinalités dans le calcul de la similarité avec une pénalité moindre. Pour illustration, considérons les deux ensembles de groupes musicaux suivants :

$$E_1 = \{ \text{« Radiohead »}, \text{« Major Lazer »} \}$$

$$E_2 = \{ \text{« Radiohead »}, \text{« System of a Down »}, \text{« La Femme »}, \text{« Major Lazer »}, \text{« Die Antwoord »}, \text{« Biffy Clyro »} \}$$

En utilisant la fonction de Jaccard, on obtient $f_{ensemble}(E_1, E_2) = 0,33$. Cependant, la fonction de Halkidi donne $f_{ensemble}(E_1, E_2) = 0,66$, ce qui nous paraît, dans notre contexte, plus raisonnable au regard de la similitude entre les deux ensembles comparés.

10.2.2 Calcul de similarité entre EN complexes

Nous présentons dans cette sous-section, notre contribution composée de quatre approches de calcul de similarité entre EN complexes. Les EN complexes traitées sont composées de n propriétés et nous rappelons que $s_i(p_{1i}, p_{2i})$ est de score de similarité entre les propriétés au rang i de o_1 et o_2 . Nos trois premières approches ciblent l'agrégation de tous les scores intermédiaires $s_i(p_{1i}, p_{2i})$ pour le calcul de la similarité globale.

- La première est basée sur la combinaison linéaire des $s_i(p_{1i}, p_{2i})$. Contrairement aux nombreuses propositions de l'état de l'art, nous utilisons une technique par étalonnage pour déterminer les poids de différentes similarités intermédiaires.
- La deuxième exploite la régression logistique pour évaluer le score de similarité entre EN complexes comme la probabilité qu'elles soient identiques, sachant les différentes similarités entre propriétés.
- La troisième exploite la technique d'apprentissage supervisé basée sur les arbres de décision pour calculer le score de similarité entre EN complexes à partir des $s_i(p_{1i}, p_{2i})$.
- La quatrième approche, quant à elle, ne s'appuie pas sur l'agrégation des $s_i(p_{1i}, p_{2i})$. Elle combine

de façon originale le clustering et le modèle vectoriel pour le calcul du score de similarité entre EN complexes.

10.2.2.1 Approche par combinaison linéaire avec pondération par étalonnage

Cette proposition reprend l'approche par combinaison linéaire pour agréger les scores de similarité entre propriétés. La similarité s correspond à la somme pondérée des s_i et elle est donnée littéralement par :

$$s(o_1, o_2) = \sum_{i=1}^n w_i \cdot s_i(p_{1i}, p_{2i}) \quad (1)$$

w_i est le poids donné à la propriété au rang i et w_1, w_2, \dots, w_n sont définis tels que :

$$\sum_{i=1}^n w_i = 1 \quad \text{et} \quad \forall 1 \leq i \leq n, w_i \geq 0 \quad (2)$$

Contrairement aux nombreux travaux de l'état de l'art où les w_i sont définis de façon empirique, nous déterminons les poids dans cette approche en expérimentant différentes combinaisons tout en respectant la contrainte fixée par l'équation (2). En effet, un jeu de validation est constitué à partir d'un ensemble de couples d'EN complexes, de façon à couvrir un grand nombre de variantes. L'expert associe à chacun de ces couples un score de similarité. Un ensemble de combinaison de w_i est généré et la formule (1) est utilisée pour évaluer le score de similarité pour les couples d'EN du jeu de validation. La combinaison des w_i qui permet de reproduire au mieux les scores de l'expert (meilleures métriques voir section 10.3.2.3) est sélectionnée pour la mise en œuvre de l'approche.

Considérons par exemple que l'on s'intéresse au calcul de similarité entre EN complexes ayant chacune deux propriétés. On suppose aussi que pour la génération des combinaisons de poids, on fait varier les valeurs de w_1 et w_2 par pas de 0,2. Pour rester conforme à l'équation (2), les combinaisons $(w_1; w_2)$ possibles sont les suivantes : (0; 1), (0, 2; 0, 8), (0, 4; 0, 6), (0, 6; 0, 4), (0, 8; 0, 2), (1; 0). Chacune de ces combinaisons est ensuite utilisée dans la formule (2) pour évaluer la similarité entre couples d'EN du jeu de validation. La combinaison permettant d'obtenir les meilleures métriques d'évaluation est utilisée pour la mise en œuvre du calcul de similarité.

10.2.2.2 Approche par régression logistique

Pour notre deuxième approche, nous voulons également traiter $s(o_1, o_2)$ comme une combinaison linéaire des $s_i(p_{1i}, p_{2i})$. Contrairement à l'approche précédente, nous proposons d'exploiter non plus l'étalonnage, mais plutôt des techniques par apprentissage pour déterminer les pondérations optimales. Nous avons envisagé d'utiliser la régression linéaire [38]. Cependant, dans notre cas, $s(o_1, o_2) \in [0, 1]$ et d'après Agresti et al. [1], « *la régression linéaire est adaptée pour le cas où la plage des valeurs à prédire couvre l'ensemble des réels. Ce modèle présente des limites lorsque ces valeurs doivent appartenir à l'intervalle [0, 1]* » (les valeurs de prédiction peuvent parfois être négatives et proches de 0, ou supérieure à 1). C'est pour cette raison que nous avons opté pour la régression logistique. Cette technique s'appuie sur la fonction *logit* [5] pour construire une bijection de $[0, 1]$ dans l'ensemble des réels.

Pour deux EN complexes o_1 et o_2 , l'approche par régression logistique évalue leur score de similarité comme la probabilité qu'elles soient identiques étant données les différentes similarités entre propriétés.

Littéralement,

$$s(o_1, o_2) = P(1|X) \quad \text{avec} \quad X = (s_1(p_{11}, p_{21}), \dots, s_n(p_{1n}, p_{2n}))$$

Pour calculer la probabilité conditionnelle correspondant au score de similarité, la régression logistique s'appuie sur l'hypothèse selon laquelle le *logit* de celle-ci s'écrit comme une combinaison linéaire des éléments du vecteur X . La fonction *logit* [5] est définie par :

$$\text{logit}(s(o_1, o_2)) = \ln\left(\frac{s(o_1, o_2)}{1 - s(o_1, o_2)}\right) = w_0 + w_1 \cdot s_1(p_{11}, p_{21}) + \dots + w_n \cdot s_n(p_{1n}, p_{2n}) \quad (3)$$

Ainsi, nous obtenons :

$$s(o_1, o_2) = \frac{1}{1 + e^{-(w_0 + w_1 \cdot s_1(p_{11}, p_{21}) + \dots + w_n \cdot s_n(p_{1n}, p_{2n}))}}$$

Tout l'enjeu de cette deuxième approche est de déterminer les poids w_0, w_1, \dots, w_n permettant de calculer le *logit* (formule (3)). Nous exploitons pour cela la technique d'apprentissage basée sur la régression linéaire puisque le *logit* prend ses valeurs dans \mathbb{R} . Nous avons constitué un jeu d'entraînement composé de m couples d'EN complexes $(o_1^1, o_2^1), \dots, (o_1^m, o_2^m)$ que nous représentons par une matrice $A \in M_{m, n+1}(\mathbb{R})$. Le coefficient $a_{i,j}$ de cette matrice correspond à la similarité au rang j pour le i^e couple (o_1^i, o_2^i) et $a_{i,0} = 1$ (constante de la combinaison linéaire). Explicitement, nous obtenons :

$$a_{i,j} = s_j(p_{1j}^i, p_{2j}^i), \text{ avec } 1 \leq i \leq m \text{ et } 1 \leq j \leq n$$

Le problème revient à trouver le vecteur de poids $W \in M_{n+1,1}(\mathbb{R})$ optimal, qui permet de reproduire les scores de l'expert sur le jeu d'entraînement donné par la matrice A . Nous avons opté pour la méthode basée sur la descente de gradient [63]. Ce choix se justifie par le fait que la descente de gradient est une méthode approximative qui garantit toujours l'obtention d'une solution. En effet, l'algorithme d'estimation du vecteur W par descente de gradient s'appuie sur des itérations successives, jusqu'à atteindre un seuil d'erreur toléré. Au début du processus, w_0, w_1, \dots, w_n sont initialisés aléatoirement et les scores obtenus pour les m couples sont comparés à ceux de l'expert pour évaluer l'erreur d'estimation. À chaque itération, les valeurs des w_i sont incrémentées ou décrémentées selon le sens de variation de l'erreur. Le nombre maximum d'itérations, l'erreur maximale tolérée et le pas de l'incrément (dégrément) sont les paramètres de l'algorithme.

10.2.2.3 Approche basée sur les arbres de décision

Pour notre troisième approche, la forme de la fonction d'agrégation n'est pas connue a priori comme c'est le cas dans les deux premières. En effet, des dépendances éventuelles peuvent exister entre similarités de propriétés. Pour des EN *entreprise*, par exemple, la similarité entre activités exercées peut être corrélée à la similarité entre produits fabriqués ou commercialisés. Cette corrélation n'est pas prise en compte dans la combinaison linéaire [66]. Par conséquent, nous proposons d'utiliser une méthode d'apprentissage supervisée employée dans les systèmes de prédiction pour l'aide à la décision : les arbres de décisions [181]. L'avantage de cette approche est qu'elle reste valide dans tous les cas (corrélations entre propriétés ou pas).

Le principe des arbres de décisions est d'analyser un ensemble d'individus (couples d'EN complexes), où chacun est caractérisé par plusieurs attributs (similarités entre propriétés), pour le segmenter de

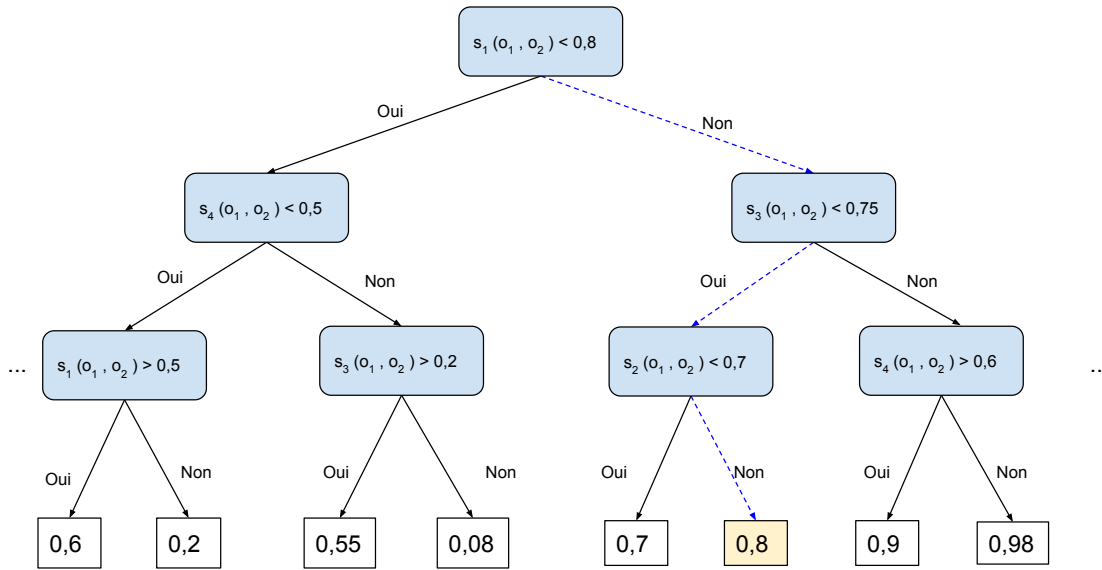


FIGURE 10.4 – Arbre de décision pour le calcul de similarité

façon optimale en sous-ensembles homogènes [159]. La segmentation se fait en posant une succession de question binaires (de type oui/non) sur les attributs de chaque individu. Le processus de segmentation des individus est représenté sous forme d’un arbre, où chaque nœud est une question portant sur un attribut donné.

Considérons qu’on s’intéresse au calcul de similarité en couples d’EN complexes (o_1, o_2) , où chacune a exactement 4 propriétés. On suppose par ailleurs qu’on sait calculer convenablement les $s_i(o_1, o_2)$. La figure 10.4 illustre un extrait de l’arbre de décision construit pour l’agrégation des similarités intermédiaires. Si, par exemple, la similarité pour la propriété au rang 1 est supérieure ou égale à 0,8, et la similarité au rang 3 est inférieure à 0,75 mais que la similarité au rang 2 est supérieure ou égale à 0,7, l’arbre de décision (chemin en pointillés) permet d’en déduire que le score de similarité global entre les deux EN comparées est de l’ordre de 0,8.

Le choix des arbres de décision plutôt qu’un autre système comme les réseaux de neurones, se justifie d’abord par le fait que ces arbres sont construits suivant une logique booléenne. D’après Harrington et al. [79], le fait que les arbres de décision soient lisibles et compréhensibles par l’humain est un facteur très important pour l’ajustement du modèle, contrairement aux réseaux de neurones, par exemple, qui sont des « boîtes noires ». De plus, le nombre de propriétés d’une EN complexe n’est généralement pas très élevé (quelques dizaines au plus), or les réseaux de neurones sont réputés efficaces lorsque les données traitées ont un grand nombre d’attributs (plusieurs milliers voir des millions).

Concernant la construction de l’arbre de décision, c’est l’algorithme ID3 [139] qui est généralement utilisée. Il est basé sur la notion de *quantité d’information* définie en théorie de l’information [167]. Nous ré-utilisons cet algorithme pour construire les arbres de décision à partir du jeu d’entraînement. Le seul paramètre que nous spécifions ici est le nombre maximal d’EN dans chaque sous-ensemble homogène.

10.2.2.4 Approche basée sur le clustering et le modèle vectoriel

Pour cette dernière approche, nous nous inspirons des travaux de Becker et al. [12] (cf. chapitre 6 de l'état de l'art), qui exploitent les méta-données embarquées dans les documents multimédia des réseaux sociaux, pour les regrouper en collections, susceptibles de décrire un même événement social. Nous nous inspirons également du modèle vectoriel de Salton [157] utilisé en RI pour apparier requêtes et documents. Notre démarche se décline en deux phases :

- un prétraitement partitionne l'ensemble des EN complexes en clusters. Pour chaque propriété, un ensemble de clusters est construit ;
- la deuxième phase exploite ce partitionnement pour évaluer la similarité entre EN. En effet, chaque EN complexe est représentée comme un vecteur dans la base constituée par l'ensemble des clusters. Nous définissons une fonction inspirée du TFIDF [156] pour calculer le poids de chaque cluster dans le vecteur. La similarité entre deux EN complexes est évaluée par le cosinus [157] de l'angle formé par les deux vecteurs les représentant.

Partitionnement de l'ensemble des EN - Pour la phase de partitionnement, nous nous inspirons de la démarche de Becker et al. [12]. Considérons un ensemble O contenant quatre EN complexes de type *événement* e_1, e_2, e_3, e_4 , chacune représentée dans un modèle à plat par le titre, la date et le lieu.

$e_1 = \langle \text{Main Square, 30 - 06 - 2017, Arras} \rangle$

$e_2 = \langle \text{Main Square Festival 2017, 30 - 06 - 2017, Citadelle - Arras} \rangle$

$e_3 = \langle \text{Florent Pagny, 13 - 10 - 2017, Zénith Toulouse} \rangle$

$e_4 = \langle \text{Florent Pagny, 22 - 09 - 2017, Zénith Arena Lille} \rangle$

Pour partitionner l'ensemble des EN, nous considérons les hypothèses suivantes :

- deux EN sont dans le même cluster de titres si et seulement si la similarité de leurs titres est supérieur à 0,7 ;
- deux EN sont dans le même cluster des dates si et seulement si la similarité de leurs dates vaut 1 ;
- deux EN sont dans le même cluster des lieux si et seulement si la similarité de leurs lieux est supérieure à 0,8.

À partir de ces hypothèses, nous regroupons les EN de O en clusters suivant les différentes propriétés. Nous obtenons en tous 8 clusters comme le montre la figure 10.5. Nous avons notamment 2 clusters de titres, 3 clusters de dates et 3 autres pour les lieux.

De façon générale, on appelle O un ensemble d'EN complexes. Pour chaque propriété p_i d'une EN complexe $o \in O$, m_i clusters $C_1^i, C_2^i, \dots, C_{m_i}^i$ sont construits. Chaque cluster contient les EN complexes pour lesquelles les propriétés au rang i sont similaires (similarité supérieure à un seuil). Cette similarité est calculée en utilisant les fonctions définies en section 10.2.1. Le cluster C_j^i correspond au j^e cluster pour la i^e propriété.

On note $\epsilon_i > 0$, le score de similarité seuil à partir duquel deux EN complexes sont considérées

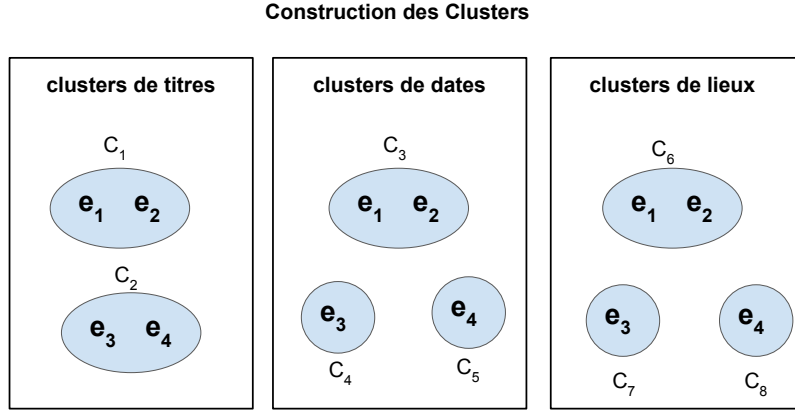


FIGURE 10.5 – Partitionnement en clusters d’un ensemble de 4 EN *événement*

similaires pour la propriété au rang i . ϵ_i est fixé de façon à garantir un partitionnement optimal des EN pour la propriété au rang i . On définit donc formellement le cluster C_j^i par :

$$C_j^i = \{o_1, o_2, \dots, o_z \mid \forall k_1, k_2, 1 \leq k_1, k_2 \leq z, k_1 \neq k_2, s_i(p_{k_1 i}, p_{k_2 i}) > \epsilon_i\}$$

Pour les n propriétés de o , on obtient dim clusters, avec $dim = \sum_{i=1}^n m_i$ ($dim = 8$ pour l’exemple précédent).

Calcul de similarité - Nous reprenons l’exemple précédent en supposant que les 8 clusters aient été construits correctement. Nous souhaitons évaluer, par exemple, la similarité des couples d’EN suivants : $(e_1, e_2), (e_3, e_4)$. Notre démarche consiste à représenter chacune de ces EN en vecteurs dans la base constituée par les 8 clusters construits précédemment. Nous considérons simplement que le poids de chaque cluster dans le vecteur vaut 1 si l’EN est dans ce dernier et 0 sinon. On obtient de ce fait les vecteurs : $e_1 = (1, 0, 1, 0, 0, 1, 0, 0)$, $e_2 = (1, 0, 1, 0, 0, 1, 0, 0)$, $e_3 = (0, 1, 0, 1, 0, 0, 1, 0)$ et $e_4 = (0, 1, 0, 0, 1, 0, 0, 1)$.

Étant donné un ensemble O d’EN complexes, dont le partitionnement a permis de construire dim clusters, nous représentons une entité complexe $o \in O$ par un vecteur des dim clusters et on note α_j^i le poids du cluster C_j^i dans ce vecteur. o est donné par :

$$o = \begin{pmatrix} C_1^1 & C_2^1 & \dots & C_{m_1}^1 & \dots & C_j^i & \dots & C_{m_n}^n \\ \alpha_1^1 & \alpha_2^1 & \dots & \alpha_{m_1}^1 & \dots & \alpha_j^i & \dots & \alpha_{m_n}^n \end{pmatrix}$$

Pour deux EN complexes, $o_1, o_2 \in O$ représentés par les vecteurs : $o_1 = (\alpha_1^1, \dots, \alpha_{m_1}^1, \dots, \alpha_j^i, \dots, \alpha_{m_n}^n)$ et $o_2 = (\beta_1^1, \dots, \beta_{m_1}^1, \dots, \beta_j^i, \dots, \beta_{m_n}^n)$, nous calculons leur score de similarité en évaluant le cosinus de l’angle orienté qu’ils forment. Littéralement, ce score correspond au rapport entre le produit scalaire des vecteurs et le produit de leurs normes euclidiennes respectives.

$$s(o_1, o_2) = \cos(o_1, o_2) = \frac{o_1 \cdot o_2}{\|o_1\| \cdot \|o_2\|} = \frac{\sum(\alpha_j^i \cdot \beta_j^i)}{\sqrt{\sum \alpha_j^{i^2}} \cdot \sqrt{\sum \beta_j^{i^2}}}$$

En reprenant l'exemple des 4 événements à plat, le calcul des normes des différents vecteurs donne :

$$\|e_1\| = \|e_2\| = \|e_3\| = \|e_4\| = \sqrt{3}$$

De même, en calculant les produits scalaires, on obtient : $e_1.e_2 = 3$ et $e_3.e_4 = 1$. Ce qui donne les scores de similarité suivants : $s(e_1, e_2) = 1$ et $s(e_3, e_4) = \frac{1}{3}$.

Concernant le calcul de α_j^i (poids du cluster C_j^i dans le vecteur représentant une EN complexe), notre proposition ne se contente pas, comme dans l'exemple, de lui donner la valeur 1 ou 0 selon que l'EN est dans le cluster ou pas. Nous avons défini une fonction de pondération dédiée, inspirée du TFIDF [156].

Soit N le nombre total d'EN dans O et $N_{C_j^i}$ le nombre d'EN dans le cluster C_j^i . On note c_j^i le centroïde de C_j^i ; c_j^i est du même type que p_i .

α_j^i est calculé par la formule suivante :

$$\alpha_j^i = \log\left(\frac{N}{N_{C_j^i}}\right) \cdot s_i(p_i, c_j^i)$$

En ce qui concerne le calcul du centroïde d'un cluster, celui-ci dépend du type de la propriété p_i à laquelle il est associé. Lorsque la propriété associée au cluster est de type :

- texte court : le centroïde est l'union des textes correspondant à la propriété. Pour le cas, du cluster C_1 de l'exemple précédent, le centroïde est représenté par « Main Square Festival 2017 » ;
- date ou heure représentée par des points sur l'axe du temps : le centroïde est le barycentre des points correspondant aux EN du cluster ;
- concept sémantique : le centroïde est donné par le plus proche ancêtre commun des concepts associés à cette propriété dans le cluster ;
- lieu : quelque soit la représentation, le centroïde est un point. En effet, nous représentons les lieux de type polygone pour cette opération par leur centre. Le centroïde du cluster correspond au barycentre de l'ensemble des points représentant ses EN. Pour l'exemple de la figure 10.5, le centroïde de C_6 est le barycentre entre le point représentant le centre de la commune d'Arras et celui associé à la *Citadelle* de la même ville.
- ensemble, comme pour les textes, le centroïde correspond à l'union des ensembles.

10.3 Application au calcul de similarité entre événements et entre représentations

Nous rappelons que le calcul de similarité entre EN complexes est utile pour deux traitements de notre architecture *Cognisearch* : l'indexation et la recherche d'information. En phase d'indexation, ce calcul permet d'intégrer sans doublon des nouvelles EN complexes dans l'index. Il est question ici, de parcourir l'index pour déterminer si l'EN à intégrer existe déjà (en partie ou de façon complète). L'enjeu étant de fusionner les informations dans le cas de l'affirmative. À cet effet, il faudrait être sûr que les deux EN fusionnées soient effectivement identiques. Par conséquent, l'évaluation de la similarité ici doit se faire en resserrant les contraintes du problème. Par exemple, certaines propriétés peuvent être requises pour définir une EN complexe. Pour une adresse par exemple, le code postal et le nom de la commune peuvent être les éléments requis.

En phase de recherche d'information, le calcul de similarité permet d'apparier les besoins d'information aux EN de l'index. Un besoin d'information peut être représenté comme une EN-requête, pour laquelle certaines propriétés peuvent être non renseignées. L'enjeu ici est de trouver toutes les EN de l'index similaires à l'EN requête pour satisfaire la demande de l'utilisateur. Contrairement au calcul de similarité en phase d'indexation, les EN retournées ici ne doivent pas forcément être identiques à l'EN requête, et le seuil de similarité requis peut-être revu. De ce fait, les contraintes du problème peuvent être relâchées pour garantir des réponses.

Nous présentons dans cette section l'expérimentation de nos différentes approches de calcul de similarité entre événements et entre représentations. Cette expérimentation est menée dans un contexte d'indexation pour lequel nous fixons des contraintes « resserrées ». La première partie pose de façon formelle notre problème, la deuxième détaille le protocole d'expérimentation et la dernière présente la mise en œuvre des approches et l'analyse des résultats obtenus.

10.3.1 Formalisation du problème

Une EN événement e selon notre modèle est définie par un titre t , une description $desc$, un ensemble de catégories Cat_e et par un ensemble de représentations R_e .

$$e = \langle t, desc, Cat_e, R_e \rangle$$

Une représentation r d'un événement e est définie par un lieu l , une date d , une heure h , un ensemble d'organismes Org_r et un ensemble d'intervenants A_r .

$$r = \langle l, d, h, Org_r, A_r \rangle, r \in R_e$$

Dans le cadre d'une première expérimentation pour les événements, nous avons choisi de nous baser sur 3 propriétés : le titre t , l'ensemble des catégories Cat_e et les intervenants principaux A_{P_e} . A_{P_e} correspondant à l'ensemble des intervenants communs à toutes les représentations.

$$A_{P_e} = \{a \mid \forall r \in R_e, a \in A_r\}$$

Pour le calcul de similarité entre représentations, nous utiliserons le lieu l , la date d et l'heure h . Ces choix se justifient par le fait que ces différentes propriétés permettent de caractériser de façon unique des représentations.

10.3.2 Protocole d'expérimentation du calcul de similarité

Nous spécifions dans cette sous-section, les fonctions utilisées pour le calcul de la similarité entre propriétés. Ensuite, nous présentons les hypothèses de travail et plus précisément les différents cas que nous souhaitons observer. Enfin, nous donnons les métriques que nous utiliserons pour évaluer les forces et les faiblesses de nos approches.

10.3.2.1 Fonctions de calcul de similarité selon le type des propriétés

Les titres d'événements sont des textes courts : nous utilisons la fonction f_{texte_court} pour évaluer leur similarité. Les catégories d'événements sont des ensembles de concepts d'ontologies, le calcul de

leur similarité est basé sur la fonction $f_{ensemble}$, où la similarité entre concepts s'appuie sur la fonction $f_{semantique}$ (Wu et Palmer). De même, le calcul de la similarité entre ensembles d'intervenants est aussi basé sur $f_{ensemble}$. Nous utilisons par contre la fonction f_{texte_court} pour l'évaluation de la similarité entre éléments de ces ensembles. Concernant la similarité entre lieux, nous utilisons la fonction f_{lieu} . Nous considérons que deux lieux de type point distant de plus de 20 km sont non similaires ($\gamma = 20$ km). Par ailleurs, pour la comparaison de deux lieux, l'un représenté par un point et l'autre par un polygone (l_2), nous définissons d (extension de communes) proportionnellement à la plus grande distance entre le centre du polygone et ses sommets. Pour nos expérimentations, nous fixons le coefficient de proportionnalité à $2/5$ comme Sallaberry et al. [154] ($d = \frac{2 \cdot \rho_{l_2}}{5}$, où ρ_{l_2} est le centroïde du polygone l_2). Enfin, nous utilisons la fonction f_{temps} pour le calcul de la similarité entre dates et heures. Pour le cas des dates, nous considérons que deux dates sont identiques si et seulement, elles correspondent au même jour ($\theta = 1$ jour). Nous considérons aussi que deux horaires avec plus de 60 minutes d'écart sont différents ($\theta = 60$ minutes).

10.3.2.2 Contexte d'expérimentation : hypothèses de travail

Lors de l'extraction d'information dans le texte des pages web, la valeur d'une ou de plusieurs propriétés d'une EN *événement* peut ne pas avoir été identifiée. Par conséquent, certaines similarités intermédiaires peuvent être inconnues. Pour calculer la similarité entre EN *événement* en phase d'indexation, nous posons comme préalable la nécessité pour les deux EN d'avoir au moins la propriété *titre* renseignée. De ce fait, un événement sans titre n'est similaire à aucun autre. Par ailleurs, nous considérons le lieu comme nécessaire pour l'évaluation de la similarité entre représentations. De ce fait, les représentations sans lieux ne sont pas considérées dans nos expérimentations.

À partir de ces deux hypothèses, nous avons constitué des familles de couples, pour tenir compte des différents cas de figure à observer dans cette expérimentation. Nous noterons une propriété manquante par un tiret (-). EN *événement* et *représentation* sont caractérisées par trois propriétés (cf. section 10.3.1). Nous définissons tout d'abord les différentes familles pour le cas des événements. Pour le cas des représentations, il suffira de remplacer le titre par le lieu, ainsi que l'ensemble des intervenants principaux et de catégories par la date et l'heure respectivement.

— **Famille 1 : toutes les propriétés sont renseignées dans chacune des deux EN**

Les couples de cette famille sont de la forme :

$$e_1 = \langle t_1, Cat_{e_1}, A_{P_{e_1}} \rangle \quad et \quad e_2 = \langle t_2, Cat_{e_2}, A_{P_{e_2}} \rangle$$

— **Famille 2 : une propriété n'est pas renseignée dans l'une des EN**

Deux cas sont possibles ici :

— la propriété manquante est l'ensemble des catégories :

$$e_1 = \langle t_1, Cat_{e_1}, A_{P_{e_1}} \rangle \quad et \quad e_2 = \langle t_2, -, A_{P_{e_2}} \rangle$$

— la propriété manquante est l'ensemble des intervenants principaux :

$$e_1 = \langle t_1, Cat_{e_1}, A_{P_{e_1}} \rangle \quad et \quad e_2 = \langle t_2, Cat_{e_2}, - \rangle$$

— **Famille 3 : la même propriété n'est pas renseignée dans chacune des deux EN**

Deux cas sont également possibles ici :

- la propriété manquante dans les deux EN est l'ensemble des catégories :

$$e_1 = \langle t_1, -, A_{P_{e_1}} \rangle \quad \text{et} \quad e_2 = \langle t_2, -, A_{P_{e_2}} \rangle$$

- la propriété manquante dans les deux EN est l'ensemble des intervenants principaux :

$$e_1 = \langle t_1, Cat_{e_1}, - \rangle \quad \text{et} \quad e_2 = \langle t_2, Cat_{e_2}, - \rangle$$

- **Famille 4 : une propriété n'est pas renseignée dans chacune des deux EN, mais ce n'est pas la même**

Dans ce cas, l'ensemble des catégories est non renseigné dans l'une des EN et l'ensemble des acteurs principaux l'est dans l'autre. Les couples de cette famille sont sous la forme suivante :

$$e_1 = \langle t_1, Cat_{e_1}, - \rangle \quad \text{et} \quad e_2 = \langle t_2, -, A_{P_{e_2}} \rangle$$

- **Famille 5 : deux propriétés ne sont pas renseignées dans au moins une des EN**

Dans cette dernière famille, une des deux EN n'a que le titre. Selon la forme de l'autre EN, plusieurs cas sont possibles :

- une EN n'a que le titre alors que l'autre est complète :

$$e_1 = \langle t_1, Cat_{e_1}, A_{P_{e_1}} \rangle \quad \text{et} \quad e_2 = \langle t_2, -, - \rangle$$

- une EN n'a que le titre et l'autre à une propriété manquante :

$$e_1 = \langle t_1, -, A_{P_{e_1}} \rangle \quad \text{et} \quad e_2 = \langle t_2, -, - \rangle$$

ou bien

$$e_1 = \langle t_1, Cat_{e_1}, - \rangle \quad \text{et} \quad e_2 = \langle t_2, -, - \rangle$$

- les deux EN n'ont qu'une seule propriété renseignée : le titre :

$$e_1 = \langle t_1, -, - \rangle \quad \text{et} \quad e_2 = \langle t_2, -, - \rangle$$

10.3.2.3 Métriques d'évaluation

Dans notre contexte, étant données deux EN complexes (événements ou représentations respectivement), l'expert n'a que deux options : elles sont similaires ou pas. Le score obtenu pour chaque couple d'EN avec nos approches est, quant à lui, comparé à un seuil pour déterminer si les EN du couple sont similaires ou pas. Les scores obtenus pour l'ensemble des couples du jeu d'évaluation sont ensuite confrontés à ceux donnés par l'expert afin de calculer la précision P , le rappel R , la F_1 -mesure et l'exactitude E (ou *Accuracy* en anglais) relatifs à chaque approche. Ces mesures sont calculées de la même façon que pour l'évaluation des systèmes de classification [40]. On note :

- **VP** : le nombre de vrai-positifs, c'est le nombre de couples d'EN considérés à la fois similaires par l'expert et par l'approche en cours d'évaluation ;
- **FP** : le nombre de faux-positifs, c'est le nombre de couples d'EN considérés similaires à tort ;

- **VN** : le nombre de vrai-négatifs, c'est le nombre de couples d'EN qui sont non similaires à la fois pour l'expert et pour l'approche en cours d'évaluation ;
 - **FN** : le nombre de faux-négatifs, c'est le nombre de couples d'EN considérés non similaires à tort.
- R , P et F_1 -mesure sont calculés selon les formules suivantes :

$$P = \frac{VP}{VP + FP}$$

$$R = \frac{VP}{VP + FN}$$

$$F_1\text{-mesure} = \frac{2.P.R}{R + P}$$

La F_1 -mesure intègre les résultats de la précision et du rappel dans une seule et même mesure. Cette mesure permet d'évaluer la capacité d'une approche à établir la similarité entre deux EN complexes. La F_1 -mesure peut toutefois s'avérer incomplète dans certaines cas. Supposons qu'on ait par exemple, $VP = 0$ et $VN = N$ (où N est le nombre d'EN non similaires d'après l'expert), on aura $P = R = 0$ et F_1 - mesure non définie. Et pourtant le fait que $VN = N$ prouve la capacité du système à établir la non-similarité entre EN. D'où l'introduction de l'exactitude E qui évalue à la fois la capacité du système à établir les similarités et les non-similarités. Elle est donnée par la formule :

$$E = \frac{VP + FN}{VP + VN + FP + FN}$$

Les F_1 -mesure et exactitude E obtenues pour les différentes approches, sont confrontées pour identifier celles qui donnent les meilleurs résultats. Dans le cadre de cette évaluation, nous utiliserons la F_1 -mesure comme métrique principale pour comparer les approches. Ce choix se justifie par le fait que notre objectif premier est d'évaluer la capacité des approches à établir la similarité entre EN complexes. Dans le cas où la différence de F_1 -mesure entre deux approches n'est pas significative (supérieure en valeur absolue à 1%), nous comparerons également les exactitudes. Nous comparons tout d'abord les différentes approches sur l'ensemble du jeu d'évaluation, ensuite, une analyse plus fine nous permettra d'observer les différentes approches selon les familles de couples d'EN. Il est important de préciser que le jeu d'évaluation est construit de façon à prendre en compte le maximum de cas de figures pour les différentes familles.

Dans cette expérimentation, nous utilisons l'approche CombMNZ normalisée [100] comme « baseline ». Le choix de CombMNZ au détriment d'autres combinateurs « classiques et simples » proposés en RI (CombAVG, CombSUM, CombMIN, CombMAX), est motivé par les résultats d'expérimentation de Lee et al. [100], ainsi que ceux de Palacio [135]. Ces expérimentations montrent que de tous des combinateurs, CombNMZ est celui qui donne les meilleurs résultats. Pour rappel, la similarité avec CombMNZ est évaluée comme la somme des similarités intermédiaires, multipliée par le nombre de ces similarités qui sont non nulles. Lee et al. [100] recommandent de normaliser les scores ainsi calculés, puisque les valeurs obtenues peuvent être supérieures à 1 selon les cas. Ainsi, pour un ensemble de couples d'EN complexes pour lesquels on a évalué les similarités s avec CombMNZ, en supposant que les valeurs obtenues varient entre s_{min} et s_{max} , le score de similarité normalisé que nous utiliserons est donné par :

$$s_{normalise} = \frac{s - s_{min}}{s_{max} - s_{min}}, \quad s_{normalise} \in [0, 1]$$

10.3.3 Expérimentations des approches de calcul de similarité

Nous présentons la mise en œuvre des différentes approches pour l'évaluation de la similarité entre EN *événement* et entre EN *représentation*. Ces deux types d'EN complexes présentent des caractéristiques différentes. Les propriétés d'EN événement dans notre contexte sont représentées par du texte court (*titre*), des ensembles de concepts d'ontologie (*catégorie*) ou encore par des ensembles de textes courts (*intervenants*). Les propriétés d'une EN *représentation*, quant à elles, sont représentées par des valeurs numériques (points, polygones, dates et heures). L'un des enjeux de cette expérimentation est d'observer le comportement de chacune de nos approches pour ces deux contextes différents. Nous appellerons dans la suite nos différentes approches respectivement approche par étalonnage, approche par régression logistique, approche basée sur les arbres de décision et approche par clustering.

10.3.3.1 Paramétrage des approches pour l'expérimentation

Concernant l'approche par étalonnage, tous les poids sont initialisés à $\frac{1}{3}$ (car nous avons trois propriétés). Les valeurs de chaque poids sont variées par pas de 0,01 sous la contrainte que la somme des poids reste égale 1. Les valeurs des poids permettant d'obtenir les meilleures mesures d'évaluation sont sélectionnées pour l'expérimentation. Afin de tenir compte des propriétés manquantes, nous assumons l'hypothèse suivante : « lorsque le score de similarité pour une propriété est inconnue, le poids de cette propriété est répartie équitablement entre les autres. » Considérons qu'on traite des EN complexes constituées de trois propriétés telles que $w_1 = 0,4$, $w_2 = 0,4$ et $w_3 = 0,2$. S'il s'avère, par exemple, que la première propriété est non renseignée pour au moins l'une des EN (s_1 est inconnue), on aura alors : $w_2 = 0,6$ et $w_3 = 0,4$.

Concernant l'approche par régression logistique, tout comme celle basée sur les arbres de décision, nous utilisons la validation croisée [7] pour déterminer les paramètres d'entraînement des modèles d'apprentissage. Par ailleurs, les familles auxquelles appartiennent les couples du jeu d'entraînement, sont également utilisées pour de la construction des modèles. L'idée est d'exploiter le maximum d'informations pour l'obtention de modèles d'apprentissage robustes et performants. Nous utiliserons le même jeu d'entraînement, comme jeu de validation de chacune des expérimentation pour la détermination des poids optimaux de l'approche par étalonnage.

En ce qui concerne l'approche basée sur le clustering, nous utilisons, comme Becker et al. [12], l'algorithme *Single Pass* [72] pour le partitionnement de l'ensemble des EN. Cet algorithme offre la capacité de construire les clusters dynamiquement en tenant compte de l'ajout de nouveaux éléments dans l'ensemble d'EN traitées. Pour le choix expérimental des seuils permettant de construire ces clusters, nous faisons varier ces valeurs entre 0,6 et 1 par pas de 0,01.

10.3.3.2 Expérimentation du calcul de la similarité entre événements

Le jeu d'évaluation ici est constitué de 100 couples d'événements couvrant les 5 familles présentées en section 10.3.2.2. Chaque famille contient des couples d'événements similaires et non similaires, pour lesquelles certaines propriétés peuvent être très proches. En effet, deux événements peuvent avoir des titres très proches, être de la même catégorie sans pour autant être similaires. C'est le cas des événements « *F1 Grand Prix De France* » et « *Grand Prix De France Moto* », qui sont de la même catégorie « *Sport Mécanique* », sans toutefois être similaires. Le tableau 10.1 détaille les proportions de chaque famille dans le jeu d'évaluation. Pour la moitié des 42 couples non similaires, les deux événements comparés

	Similaires	Non similaires	Total
Famille 1	14	9	23
Famille 2	14	8	22
Famille 3	14	9	23
Famille 4	6	7	13
Famille 5	10	9	19
Total	58	42	100

TABLE 10.1 – Jeu d'évaluation du calcul de similarité entre événements

présentent parfois des propriétés communes ou proches (mêmes catégories, mêmes intervenants, ou des titres similaires).

Nous avons mis en œuvre le test de sphéricité de Barlett [10] pour vérifier l'indépendance globale entre similarités intermédiaires sur notre jeu d'évaluation. À cet effet, nous avons utilisé l'implémentation du test de Barlett de la librairie Python *scipy*⁹⁵. Le jeu de données pour ce test est constitué de 50 couples d'événements, pour lesquels toutes les propriétés sont renseignées. L'hypothèse nulle⁹⁶ est confirmée avec un niveau de confiance à 95%. On en déduit donc que les différentes similarités entre propriétés sont globalement indépendantes. Par conséquent, expérimenter les approches basées sur la combinaison linéaire des similarités intermédiaires sur notre jeu de données garde tout son sens.

Réglage des approches - Concernant l'approche par étalonnage, les résultats optimaux ont été obtenus pour : $w_t = 0,5$, $w_{Cat} = 0,3$ et $w_{A_{pe}} = 0,2$. Pour la régression logistique, nous avons obtenu le modèle idéal au bout de 3000 itérations avec un taux d'apprentissage à 10^{-3} et une erreur maximale tolérée à 10^{-4} . Pour l'approche basée sur les arbres de décision, nous avons constaté une stabilité de la F_1 -mesure et de l'exactitude, à partir de 10 nœuds au maximum dans chaque sous-ensemble homogène. Le jeu d'entraînement pour les approches par régression logistique et par régression basée sur les arbres de décision est le même, et contient 85 couples d'événements. Parmi ceux-ci, 50 sont similaires selon l'expert et les 35 autres ne le sont pas. Ce jeu d'entraînement a été préparé de façon à couvrir des différentes familles d'EN. Il est important de préciser que ce jeu d'entraînement est différent du jeu d'évaluation utilisé pour le calcul des métriques. Enfin, pour l'approche par clustering, les seuils de partitionnement sont les suivants : $\epsilon_t = 0,8$, $\epsilon_{Cat} = 0,5$ et $\epsilon_{A_{pe}} = 0,6$. Pour cette dernière approche, nous avons construit les clusters à partir d'un ensemble de 300 événements. 161 clusters en tout ont été construits, dont 102 sont relatifs au titre, 18 aux catégories et 41 aux intervenants principaux.

Choix des seuils - Pour déterminer les seuils permettant de conclure de la similarité ou non de deux événements, nous avons opté pour une démarche en deux étapes.

- Une première étape représente sur le même graphique, les scores de similarité de l'expert et ceux de chaque approche pour l'ensemble des couples du jeu d'évaluation. Les figures 10.6, 10.7, 10.8, 10.9, 10.10 présentent les graphiques obtenus pour chacune des approches. Ces couples sont rangés dans l'ordre croissant des scores de l'expert. Cette première phase laisse entrevoir un seuil entre **0,6 et 0,8** pour toutes les approches.
- Une deuxième étape explore l'intervalle contenant le seuil $([0,6;1])$ pour déterminer celui qui

95. <https://www.scipy.org/>

96. https://www.math.u-psud.fr/pansu/web_ifips/Tests.pdf

optimise les résultats (F_1 -mesure et exactitude). L'exploration part de la valeur minimale de cet intervalle et le parcourt en entier, en incrémentant à chaque fois la valeur du seuil de **0,01**.

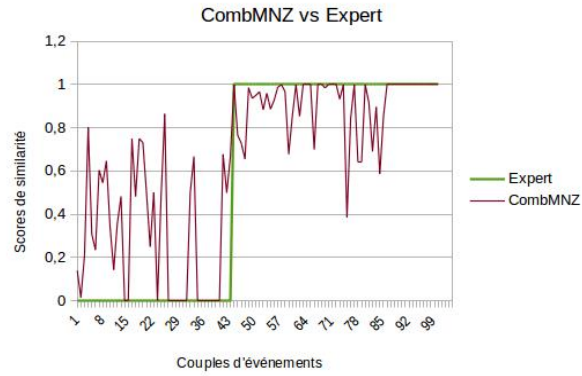


FIGURE 10.6 – Variations entre scores de similarité de l'expert et ceux de CombMNZ

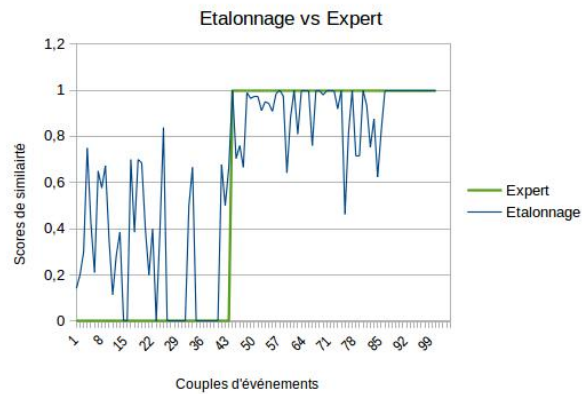


FIGURE 10.7 – Variations entre scores de similarité de l'expert et ceux calculés par l'approche par étalonnage



FIGURE 10.8 – Variations entre scores de similarité de l'expert et ceux calculés par régression logistique

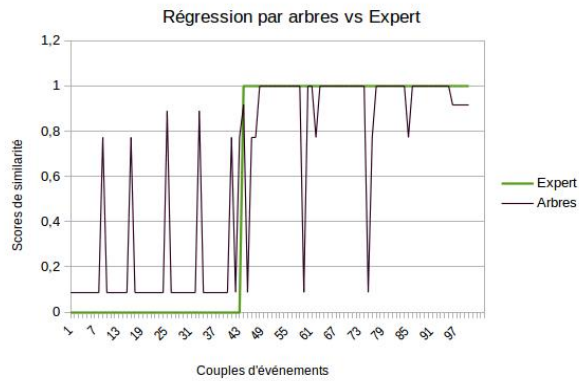


FIGURE 10.9 – Variations entre scores de similarité de l’expert et ceux calculés en exploitant les arbres de décision

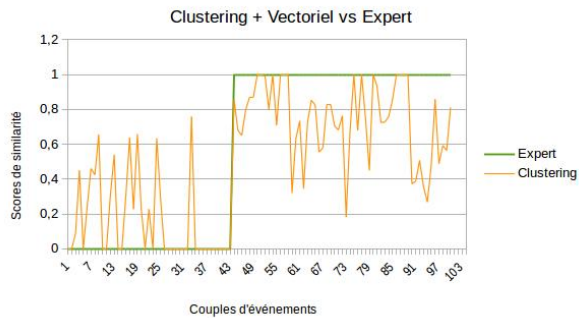


FIGURE 10.10 – Variations entre scores de similarité de l’expert et ceux calculés en combinant clustering et modèle vectoriel

Pour la similarité entre événements, les seuils obtenus pour chaque approche sont contenus dans le tableau 10.2.

Approches	Seuils optimaux
CombMNZ	0,64
Etalonnage	0,62
Régression Logistique	0,75
Régression par arbres	0,77
Clustering	0,64

TABLE 10.2 – Seuils optimaux pour la similarité entre événements

Résultats et analyses - Le tableau 10.3 présente les résultats obtenus sur l’ensemble des 100 couples d’événements du jeu d’évaluation. Nous pouvons constater que les deux approches basées sur l’apprentissage (régression logistique et arbres de décision) se démarquent clairement des autres en termes de F_1 -mesure et d’exactitude. De ces deux approches, c’est la régression logistique qui donne ici les meilleurs

Approches	CombMNZ	Étalonnage	Rég. logistique	Rég. par arbres	Clustering
Précision	79,03	79,03	89,09	84,48	84,91
Rappel	84,48	86,21	84,48	84,48	76,27
F ₁ -mesure	81,67	82,64	86,73	84,48	80,36
Exactitude	78,00	79,00	85,00	82,00	78,00

TABLE 10.3 – Résultats obtenus avec les différentes approches sur l’ensemble des couples d’événements du jeu d’évaluation

résultats. En effet, l’exploitation des familles de couples en phase d’apprentissage favorise la prise de décision lorsque certaines propriétés ne sont pas renseignées. L’approche par clustering est celle qui donne la F₁-mesure et l’exactitude les plus faibles sur l’ensemble du jeu d’évaluation. Une analyse plus fine suivant les différentes familles permettra d’expliquer ces résultats. L’approche par étalonnage donne un très bon rappel ce qui montre sa capacité à établir la similarité entre EN effectivement similaires. En effet, lorsque deux événements sont similaires, leurs titres sont le plus souvent proches. Par conséquent, même si les autres propriétés ne sont pas renseignées, le processus de calcul de similarité avec l’approche par étalonnage donnera une valeur supérieure au seuil (le poids du titre est le plus important). Le revers étant que, lorsque les titres sont proches et pourtant les événements ne sont pas similaires, cette approche aura tendance à dire le contraire : ce qui explique la faible précision.

Nous allons maintenant nous intéresser à l’analyse des résultats suivant les différentes familles de couples. Les métriques calculées pour chacune d’elles sont présentées dans le tableau 10.4. Concernant les couples d’EN *événement* pour lesquelles toutes les propriétés sont renseignées (famille 1), les meilleurs résultats sont obtenus par l’approche par clustering. C’est également l’approche qui donne les meilleurs résultats lorsque les deux EN comparées ont la même propriété non renseignée (famille 3). Ceci s’explique par le fait que lorsque deux EN sont similaires, les vecteurs les représentant le sont également et donc l’angle entre eux est faible. Cependant, dans le cas où on a des propriétés non renseignées et qu’elles ne soient pas les mêmes (famille 2, 4 et 5), nous constatons que l’approche par clustering donne les résultats les moins bons. En effet, lorsqu’une propriété est absente, tous les clusters qui lui sont associés ont un poids de 0 dans la représentation vectorielle de l’EN. De ce fait, le produit scalaire est faible par rapport au produit des normes des vecteurs : d’où le score inférieur au seuil. Pour illustration, considérons deux événements similaires e_1 et e_2 avec des titres proches ; il manque la catégorie pour le premier et la liste des intervenants principaux pour le deuxième. On considère par ailleurs qu’on ait 3 clusters de titres, 3 clusters de catégories et 3 clusters d’intervenants principaux. Les deux vecteurs représentant ces EN dans la base des clusters sont donnés par :

$$\begin{aligned} e_1 &= (0,75 \quad 0,31 \quad 0,181 \quad 0 \quad 0 \quad 0 \quad 0,75 \quad 0,2 \quad 0,15) \\ e_2 &= (0,70 \quad 0,31 \quad 0,181 \quad 0,42 \quad 0,82 \quad 0,05 \quad 0 \quad 0 \quad 0) \end{aligned}$$

Le produit scalaire entre ces deux vecteurs vaut 0,65, et leurs normes sont respectivement de 1,14 et 1,21. Ainsi le cosinus entre ces deux vecteurs vaut 0,47, ce qui est en deçà de 0,64 fixé comme seuil. D’où le rappel moyen inférieur à 70% obtenu avec cette approche dans le cas où l’on n’a pas la même propriété non renseignée dans les EN comparées

Lorsque toutes les propriétés sont renseignées (famille 1) les approches CombMNZ et par étalonnage permettent d’obtenir le même rappel. En effet, CombMNZ est un cas particulier de l’approche par éta-

Approches	CombMNZ	Étalonnage	Rég. logistique	Rég. par arbres	Clustering
$\mathbf{P}_{Famille_1}$	92,86	86,67	100	92,86	93,33
$\mathbf{R}_{Famille_1}$	92,86	92,86	85,71	92,86	100
$\mathbf{F}_{1Famille_1}$	92,86	89,66	92,31	92,86	96,55
$\mathbf{E}_{Famille_1}$	91,30	86,96	91,30	91,30	95,65
$\mathbf{P}_{Famille_2}$	85,71	92,31	86,67	91,67	85,71
$\mathbf{R}_{Famille_2}$	85,71	85,71	92,86	78,57	80,00
$\mathbf{F}_{1Famille_2}$	85,71	88,89	89,66	84,62	82,76
$\mathbf{E}_{Famille_2}$	81,81	86,36	86,36	81,81	77,27
$\mathbf{P}_{Famille_3}$	80,00	85,71	85,71	76,47	86,67
$\mathbf{R}_{Famille_3}$	85,71	85,71	85,71	92,86	92,86
$\mathbf{F}_{1Famille_3}$	82,76	85,71	85,71	83,87	89,66
$\mathbf{E}_{Famille_3}$	78,26	82,61	82,61	78,26	86,96
$\mathbf{P}_{Famille_4}$	62,50	62,50	83,33	83,33	75,00
$\mathbf{R}_{Famille_4}$	83,33	83,33	83,33	83,33	50,00
$\mathbf{F}_{1Famille_4}$	71,43	71,43	83,33	83,33	60,00
$\mathbf{E}_{Famille_4}$	69,23	69,23	84,62	84,62	69,23
$\mathbf{P}_{Famille_5}$	63,64	61,54	87,50	77,78	60
$\mathbf{R}_{Famille_5}$	70,00	80,00	70,00	70,00	30
$\mathbf{F}_{1Famille_5}$	66,67	69,57	77,78	73,68	40
$\mathbf{E}_{Famille_5}$	63,16	63,16	78,95	73,68	52,63

TABLE 10.4 – Résultats obtenus avec les différentes approches sur les couples d'événements des différentes familles

lonnage où tous les poids sont égaux et valent $\frac{1}{3}$. De ce fait, s'il y a similarité entre EN, les scores obtenus avec ces deux approches sont « équivalents », d'où les rappels similaires. Par contre, comme l'étalonnage se fait sur l'ensemble des familles et que celle-ci donne plus d'importance au titre et aux catégories (poids de 0,5 et 0,3 respectivement), si deux événements non similaires ont des titres et des catégories proches mais des acteurs principaux différents, alors l'approche par étalonnage aura tendance à donner un score au dessus du seuil à tort. Pour ce même cas, CombMNZ associe aux similarités de toutes les propriétés le même poids, ce qui permet de relativiser la similarité des deux autres propriétés d'où la meilleure précision. Lorsqu'une propriété n'est pas renseignée dans chacune des EN et que ce n'est pas la même (famille 4) ou bien lorsqu'une des EN au moins ne contient que le titre (famille 5), les approches CombMNZ et par étalonnage n'utilisent que le titre pour évaluer la similarité, ce qui impacte la précision pour les cas où les titres sont similaires alors que les événements ne le sont pas.

Enfin, dans le cas où les EN comparées ont au moins une propriété manquante, mais que ce n'est pas la même dans les deux, les résultats obtenus avec les approches basées sur l'apprentissage sont les meilleurs. Ceci s'explique par le fait qu'elles exploitent les caractéristiques des familles des couples en plus des similarités intermédiaires pour le calcul de la similarité globale.

Conclusion - Pour le calcul de similarité entre EN *événement*, nous constatons que l'approche par clustering donne les meilleurs résultats lorsque toutes les propriétés des deux EN sont renseignées (famille 1). Nous constatons aussi que dans le cas où au moins une de ces EN a une ou plusieurs propriétés non renseignées, c'est l'approche par régression logistique qui donne les meilleurs résultats dans notre contexte. Ce dernier résultat est confirmé par la tableau 10.5, synthèse des résultats pour les familles 2, 3, 4 et 5. Toutefois, pour le cas particulier des couples de la famille 3 (chaque événement a une propriété manquante

et c'est la même dans les deux), l'approche par clustering donne de meilleurs résultats.

Approches	CombMNZ	Etalonnage	Rég. logistique	Rég. par arbres	Clustering
$\mathbf{P}_{PropManq}$	75,00	77,08	86,05	81,82	81,85
$\mathbf{R}_{PropManq}$	81,82	84,09	84,09	81,82	68,89
$\mathbf{F}_{1PropManq}$	78,26	80,43	85,06	81,82	74,70
$\mathbf{E}_{PropManq}$	74,03	76,62	83,12	79,22	72,73

TABLE 10.5 – Résultats obtenus avec les différentes approches sur les couples d'événements partiellement renseignés

10.3.3.3 Expérimentation du calcul de la similarité entre représentations

Comme pour l'expérimentation du calcul de similarité entre événements, le corpus d'évaluation ici est aussi constitué de 100 couples de représentations. Rappelons qu'une représentation pour notre problème est caractérisée par un lieu, une date et une heure. Toutes les familles détaillées en section 10.3.2.2 sont représentées et le tableau 10.6 présente leur répartition. Le jeu d'évaluation est varié et contient notamment des représentations pour lesquelles les lieux correspondent à des points de l'espace ou à des polygones. Le cas des représentations ayant lieu au même endroit, à la même date et à des horaires différents sont également pris en compte. À la différence des entités événements, toutes les propriétés d'une représentation sont des valeurs numériques (polygones et de points de l'espace pour les lieux, points de l'axe du temps pour les dates et les heures).

Nous avons également mis en œuvre le test de Bartlett [10] sur le jeu d'évaluation concernant les représentations. Le test a permis de confirmer, avec un niveau de confiance à 95%, l'indépendance globale entre similarité des propriétés d'EN *représentation*. Ce qui légitime l'expérimentation des approches basées sur combinaison linéaire avec ce jeu de données.

	Similaires	Non similaires	Total
Famille 1	12	9	21
Famille 2	12	9	21
Famille 3	11	10	21
Famille 4	13	6	19
Famille 5	10	8	18
Total	58	42	100

TABLE 10.6 – Jeu d'évaluation du calcul de similarité entre représentations

Réglage des approches - L'approche par étalonnage est optimale pour : $w_l = 0,31$, $w_d = 0,36$ et $w_h = 0,33$. Concernant la régression logistique, le modèle optimal est obtenu au bout de 2000 itérations pour un pas d'apprentissage de 10^{-3} et une erreur maximale à 10^{-4} . Le modèle optimal pour l'approche basée sur les arbres de décisions est obtenu pour un nombre de nœuds maximal par sous-ensemble homogène de 6. Dans ces deux derniers cas, le jeu d'entraînement est constitué de 90 couples de représentations : 52 sont similaires selon le jugement de l'expert et 38 ne le sont pas. Comme pour les événements, les

jeux d'entraînement et d'évaluation sont différents. Concernant la construction des clusters, les seuils de partitionnement que nous avons utilisés sont les suivants : $\epsilon_l = 0,80$, $\epsilon_d = 1$ et $\epsilon_h = 0,85$. L'ensemble des représentations est constitué de 350 EN. 77 clusters ont été construits à partir de cet ensemble, parmi lesquels, 36 clusters de lieux, 28 clusters de dates et 13 clusters d'heures.

Choix de seuils - Les seuils de similarité pour les représentations sont déterminés de la même façon que pour les événements. L'observation des figures 10.11, 10.12, 10.13, 10.14 laisse entrevoir un seuil entre **0,6** et **0,8** pour les 4 approches correspondantes. Pour l'approche par clustering, le seuil pourrait être inférieur à 0,6. Dans tous les cas, il est entre **0,4** et **0,8**. Nous explorons ensuite les valeurs des intervalles supposés contenir les seuils par pas de 0,01. Les valeurs permettant d'optimiser les résultats (F_1 -mesure et exactitude) sur notre jeu d'évaluation sont répertoriées dans le tableau 10.7.

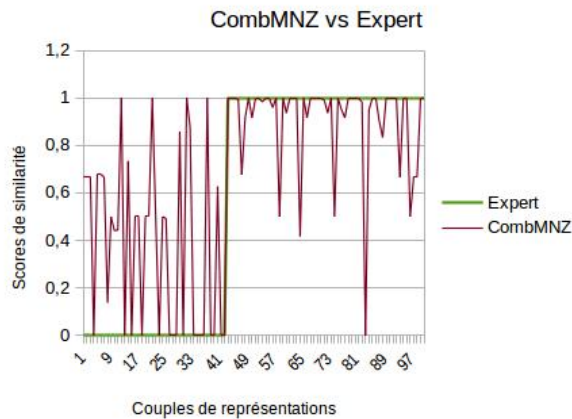


FIGURE 10.11 – Variations entre scores de similarité de l'expert et ceux de CombMNZ

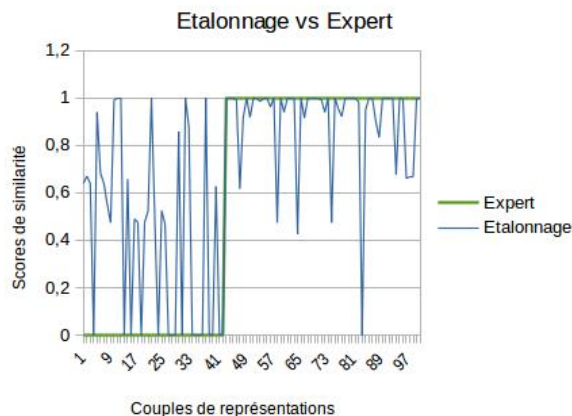


FIGURE 10.12 – Variations entre scores de similarité de l'expert et ceux calculés par l'approche par étalonnage



FIGURE 10.13 – Variations entre scores de similarité de l'expert et ceux calculés par régression logistique

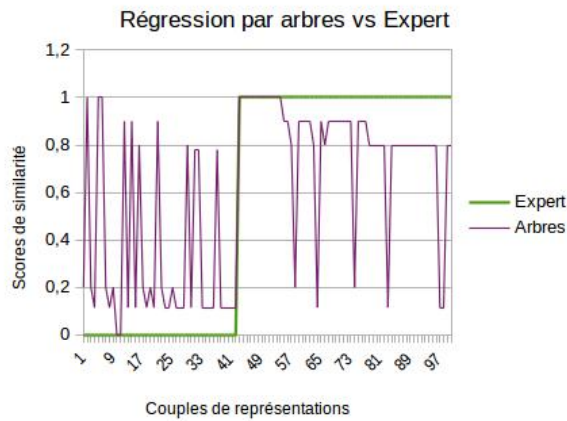


FIGURE 10.14 – Variations entre scores de similarité de l'expert et ceux calculés en exploitant les arbres de décision

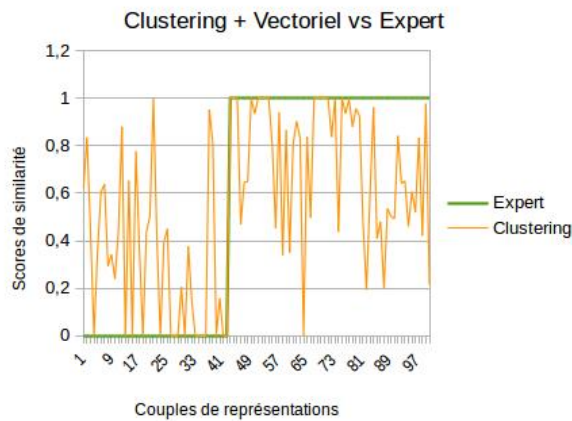


FIGURE 10.15 – Variations entre scores de similarité de l'expert et ceux calculés en combinant clustering et modèle vectoriel

Approches	Seuils optimaux
CombMNZ	0,67
Étalonnage	0,64
Régression Logistique	0,68
Régression par arbres	0,79
Clustering	0,63

TABLE 10.7 – Seuils optimaux pour la similarité entre représentations

Approches	CombMNZ	Étalonnage	Rég. logistique	Rég. par arbres	Clustering
Précision	82,76	77,59	86,21	84,48	86,28
Rappel	84,21	88,24	86,21	85,22	69,84
F₁-mesure	83,48	82,57	86,21	85,22	77,19
Exactitude	81,00	81,00	84,00	83,00	74,00

TABLE 10.8 – Résultats obtenus avec les différentes approches sur l'ensemble des couples de représentations du jeu d'évaluation

Résultats et analyses - Les résultats de l'expérimentation du calcul de similarité pour les 100 couples de représentations du jeu d'évaluation sont présentés dans le tableau 10.8. Nous constatons que, comme pour les événements, les deux approches basées sur l'apprentissage donnent les meilleurs résultats dans l'ensemble. L'approche par étalonnage permet d'obtenir le meilleur rappel, ce qui confirme sa capacité à retrouver, comme pour les événements un maximum de couples de représentations similaires (avec beaucoup de bruit comme le montre la précision). L'approche par clustering permet d'obtenir la meilleure précision, par contre c'est l'approche qui donne les moins bons résultats pour la F₁-mesure et l'exactitude. Nous allons maintenant analyser les résultats obtenus par familles de couples.

En ce qui concerne les couples d'EN pour lesquelles toutes les propriétés sont renseignées (famille 1), les approches CombMNZ, par étalonnage et par clustering sont ex-aequo. En effet, pour le cas des représentations, toutes les propriétés sont de type numérique. Ainsi, pour deux représentations identiques avec toutes les propriétés renseignées, les valeurs de celles-ci coïncident ou elles sont très proches. Ce qui explique la capacité de ces approches à établir clairement la similarité (rappel à 100%). Cependant, deux représentations distinctes se tenant le même jour, à la même heure, et à des lieux très voisins (Théâtre des Champs-Élysées et Tour Eiffel) sont considérées comme similaires à tort par toutes ces approches (précision différente de 100%).

Pour les couples ayant une même propriété non renseignée dans chacune des EN (famille 3), l'approche par clustering donne les meilleurs résultats pour le rappel, la F₁-mesure et l'exactitude. Considérons par exemple que la date soit la seule propriété non renseignée, si les deux représentations sont similaires, alors ils ont forcément des lieux et des heures très proches. Par conséquent, l'angle entre les vecteurs associés est très faible, d'où le bon rappel obtenu. Cependant, lieux et heures peuvent aussi être similaires sans que les deux représentations ne le soient pour autant. Dans ce dernier cas, l'approche par clustering donnera une score élevé à tort, ce qui impacte la précision. Notons que pour ce dernier cas, presque toutes nos approches manquent d'informations pour établir la non-similarité. Seules les approches par apprentissage exploitent la famille d'appartenance du couple pour trancher.

Lorsqu'on a au moins une propriété manquante dans au moins une des EN sans pour autant que ce ne soit la même (familles 2, 3 et 5), les deux approches par apprentissage (régression logistique et arbres

Approches	CombMNZ	Etalonnage	Rég. logistique	Rég. par arbres	Clustering
$P_{Famille_1}$	91,67	91,67	91,67	91,67	91,67
$R_{Famille_1}$	100	100	91,67	91,67	100
$F_1_{Famille_1}$	95,56	95,56	91,67	91,67	95,56
$E_{Famille_1}$	95,24	95,24	90,48	90,48	95,24
$P_{Famille_2}$	83,33	75	83,33	83,33	91,67
$R_{Famille_2}$	83,33	90	90,91	91,91	68,75
$F_1_{Famille_2}$	83,33	81,82	86,96	86,96	78,57
$E_{Famille_2}$	80,95	80,95	85,71	85,71	71,43
$P_{Famille_3}$	81,82	72,73	81,82	90,91	81,82
$R_{Famille_3}$	81,82	88,89	90	76,92	100
$F_1_{Famille_3}$	81,82	80	85,71	83,33	90
$E_{Famille_3}$	80,95	80,95	85,71	80,95	90,48
$P_{Famille_4}$	76,92	76,92	84,62	76,92	83,33
$R_{Famille_4}$	83,33	83,33	84,62	90,91	41,67
$F_1_{Famille_4}$	80	80	84,62	83,33	55,56
$E_{Famille_4}$	73,68	73,68	78,95	78,95	57,89
$P_{Famille_5}$	72,73	77,78	80,00	80,00	80,00
$R_{Famille_5}$	80,00	70,00	90,00	80,00	53,33
$F_1_{Famille_5}$	76,19	73,68	81,82	80,00	64,00
$E_{Famille_5}$	72,22	72,72	77,78	77,78	50,00

TABLE 10.9 – Résultats obtenus avec les différentes approches sur les couples de représentations des différentes familles

de décision) se démarquent des autres et la régression logistique donne les meilleurs résultats. En effet, pour ces couples, les autres approches utilisent la ou les seule(s) propriété(s) renseignées pour établir la similarité. De ce fait, lorsque les deux EN ne sont pas similaires et que les propriétés renseignées le sont, elles auront tendance à les considérer similaires à tort. Par contre, les deux approches basées sur l'apprentissage exploitent en plus l'information relative aux familles de couples pour prendre les décisions. Concernant l'approche basée sur le clustering, on remarque que pour les couples des familles 2, 4 et 5, les rappels sont faibles par rapport aux précisions obtenues. En fait, lorsque ce n'est pas la même propriété qui est non renseignée, les deux vecteurs contiennent une succession de 0 à des positions différentes. Par conséquent, le produit scalaire des deux vecteurs est faible au regard du produit des normes. Ainsi, pour deux représentations potentiellement similaires, on obtient un score en deçà du seuil et le système les considère non similaires à tort.

Conclusion - En somme, pour le calcul de similarité entre EN *représentations*, les approches CombMNZ, par étalonnage et par clustering permettent d'obtenir les mêmes résultats lorsque les propriétés des deux EN sont toutes renseignées (couples de la famille 1). Le tableau 10.10 présente la synthèse des résultats pour le cas où au moins une propriété est non renseignée. Nous pouvons constater que l'approche par régression logistique donne les meilleurs résultats en termes de F_1 -mesure et d'exactitude. Elle est talonnée de près par l'approche basée sur les arbres de décision. Pour le cas particulier où une seule et même propriété est non renseignée dans les deux représentations (famille 3), l'approche par clustering est la plus adaptée. Ces résultats confirment ceux obtenus pour l'expérimentation du calcul de similarité entre événements.

Approches	CombMNZ	Étalonnage	Rég. logistique	Rég. par arbres	Clustering
$\mathbf{P}_{PropManq}$	80,43	73,91	84,78	82,61	84,62
$\mathbf{R}_{PropManq}$	80,43	85	84,78	84,44	63,46
$\mathbf{F}_{1PropManq}$	80,43	79,07	84,78	83,52	72,53
$\mathbf{E}_{PropManq}$	77,22	77,22	82,28	81,01	68,35

TABLE 10.10 – Résultats obtenus avec les différentes approches sur les les couples de représentations partiellement renseignés

10.4 Bilan

Ce chapitre a porté sur le calcul de similarité entre EN complexes. Dans les travaux de l'état de l'art, nous avons vu que ce calcul se décline habituellement en deux phases : (i) la première phase évalue la similarité entre les valeurs de propriétés au même rang (similarité intermédiaire); (ii) et, la deuxième agrège les scores de similarité obtenus à la première phase. Concernant le calcul de similarité entre propriétés, nous avons repris des métriques proposées dans l'état de l'art. Pour le cas particulier de la similarité entre lieux, l'un représenté par une polygone et l'autre par un point de l'espace, nous avons proposé une nouvelle formule de calcul de similarité qui complète celle de l'état de l'art [152]. Cette proposition tient compte du fait que le point puisse se trouver dans le voisinage du polygone, sans forcément y être inclus.

En ce qui concerne l'agrégation des scores de similarité intermédiaires, nous formulons trois propositions :

- la première ré-utilise la combinaison linéaire pour laquelle les poids des similarités intermédiaires sont déterminés par étalonnage ;
- la deuxième exploite la régression logistique sur un jeu d'apprentissage, pour construire un modèle permettant d'évaluer la similarité globale à partir des similarités entre propriétés ;
- la dernière utilise la régression basée sur les arbres de décisions pour inférer un modèle d'apprentissage, permettant de déterminer le score de similarité global à partir de ceux des différentes propriétés.

Nous proposons également une quatrième approche qui n'est pas basée sur l'agrégation des scores de similarité entre propriétés. Cette proposition exploite les travaux de Becker et al. [12], pour partitionner l'ensemble des EN en clusters. Ensuite, chaque EN complexe est représentée comme un vecteur dans la base constituée par l'ensemble des clusters. Enfin, le modèle vectoriel de Salton [157] est utilisé pour calculer la similarité entre ces entités, en évaluant le cosinus de l'angle formé par leurs vecteurs.

Nous expérimentons toutes ces approches pour le calcul de similarité entre événements et entre représentations en phase d'indexation. Ces deux contextes d'évaluation sont différents dans la mesure où, pour le cas des événements, les propriétés peuvent être du texte, des ensembles de textes, ou des ensembles de concepts d'ontologie, alors que pour les représentations, toutes les propriétés sont de type numérique : points de l'espace, polygones ou encore points de l'axe du temps. Un premier prototype du service *Cognisearch Event* (détaillé au chapitre 11) a permis d'extraire 8 000 EN *événement* dans un corpus de 23 000 pages web de sites de billetterie. Nous avons constaté que plus de 73% des parties « *événement* » de ces EN avaient au moins une propriété non renseignée, de même que plus de 56% des parties « *représentation* ». Ce qui justifie l'importance de l'analyse des résultats de nos approches par famille de couples. Nous constatons que, lorsque les EN sont complètement renseignées, le clustering com-

biné au modèle vectoriel donne les meilleurs résultats. Par contre ses résultats sont moins bons dès que des propriétés sont non renseignées. Notons par ailleurs que les approches CombMNZ et par étalonnage donnent aussi de bons résultats lorsque toutes les propriétés sont renseignées. Cependant, lorsque des propriétés sont non renseignées, l'approche par régression logistique donne de meilleurs résultats pour nos deux expérimentations.

Approches	<i>P</i> en %	<i>R</i> en %	<i>F</i> ₁ – mesure en %
Nikolov et al. [131]	94	74	83
Khrouf et al. [89]	88	96	92
Clustering	93,33	100	96,55

TABLE 10.11 – Résultats des expérimentations de Nikolov et al. [131] et Khrouf et al. [89]

Nous avons comparé nos résultats avec ceux obtenus dans certaines travaux de l'état de l'art (tableau 10.11). Il s'agit notamment des travaux de Nikolov et al. [131] et ceux de Khrouf et al. [89]. Ces évaluations sont réalisées dans un contexte où toutes les propriétés sont renseignées. Pour un tel contexte, nos résultats sont dans le même ordre de grandeur que ceux de Khrouf et al. dont l'objectif est proche du nôtre : agréger des données d'événements provenant de plusieurs sources. De même, nos précisions pour le cas des événements et des représentations sont proches de ceux de Nikolov et al. Par contre, nos rappels sont clairement meilleurs. Ceci s'explique par le fait, qu'ils ciblent l'appariement d'enregistrements issus de bases de données différentes et leur stratégie de calcul de similarité vise à maximiser la précision, quitte à pénaliser le rappel. Pour leur contexte de travail, un appariement erroné est moins tolérable qu'un appariement manqué.

Il n'existe pas, à notre connaissance, de travaux ciblant le calcul de similarité entre EN complexes dont les propriétés sont partiellement renseignées. Nos propositions ont été jugées originales dans la communauté scientifique qui traite des informations géographiques. Nos travaux ont été validés par des publications dans les actes de l'atelier EXCES [57] de la conférence SAGEO 2017, ainsi que ceux de l'atelier international GIR 2017 [60]. Nous avons mis à disposition de la communauté le code source de l'implémentation de nos approches de calcul de similarité ainsi nos jeux de données d'expérimentation (couples d'EN *événement* et *représentation*). Ces ressources sont disponibles sur *github*⁹⁷.

97. <https://github.com/armelCPU>

Chapitre 11

Mise en œuvre de l’architecture

Cognisearch

Nous avons développé une première version des deux prototypes de services *Cognisearch Business* et *Cognisearch Event*. L’objectif visé à travers ceux-ci est de vérifier la faisabilité de l’architecture que nous avons conçue et d’expérimenter l’implémentation de nos contributions. Dans ce chapitre, nous décrivons le fonctionnement global de cette architecture en mettant l’accent sur les modules de la chaîne que nous n’avons pas encore détaillés. Nous présentons aussi l’implémentation de chaque module avec les technologies mobilisées.

Pour ces implémentations, nous ré-utiliserons certains outils exploités par les deux équipes au sein desquelles se déroule la thèse : *T2I* et *Cogniteev*. Cette ré-utilisation a pour but de capitaliser au maximum le retour d’expérience emmagasiné et l’expertise de ces équipes. En effet, les chaînes d’extraction d’information mises au point par l’équipe *T2I* sont implémentées via la plateforme *GATE*⁹⁸. De même, celles développées par *Cogniteev* sont basées sur les bibliothèques du groupe de recherche de l’université de Stanford : *Stanford NLP*⁹⁹. Nous ré-utilisons ces technologies pour la mise en œuvre de nos chaînes d’extraction. En ce qui concerne l’indexation et la RI, *Cogniteev* utilise l’outil *Elasticsearch*¹⁰⁰. C’est un outil qui a l’avantage de traiter des gros volumes de données sur une architecture distribuée. Pour nos prototypes, nous utiliserons également *Elasticsearch* pour le stockage des index et la RI.

11.1 Cognisearch Business

Il s’agit d’un service de recherche géolocalisée ciblant les entreprises sur le web. Pour construire ce service, nous exploitons les données administratives d’entreprises, mises en ligne en mode open-data et le web, comme points d’entrée. L’architecture du service *Cognisearch Business* est constituée de cinq modules comme l’illustre la figure 11.1. Un premier module (filtrage du web) exploite les données d’immatriculation d’entreprises provenant de la plateforme open-data du registre du commerce comme paramètres dans la phase d’identification de leurs sites web. Un deuxième module (extraction d’information) analyse le contenu de ces sites en instanciant notre chaîne générique d’extraction d’EN complexes pour annoter

98. <https://gate.ac.uk>

99. <https://nlp.stanford.edu/>

100. <https://www.elastic.co/>

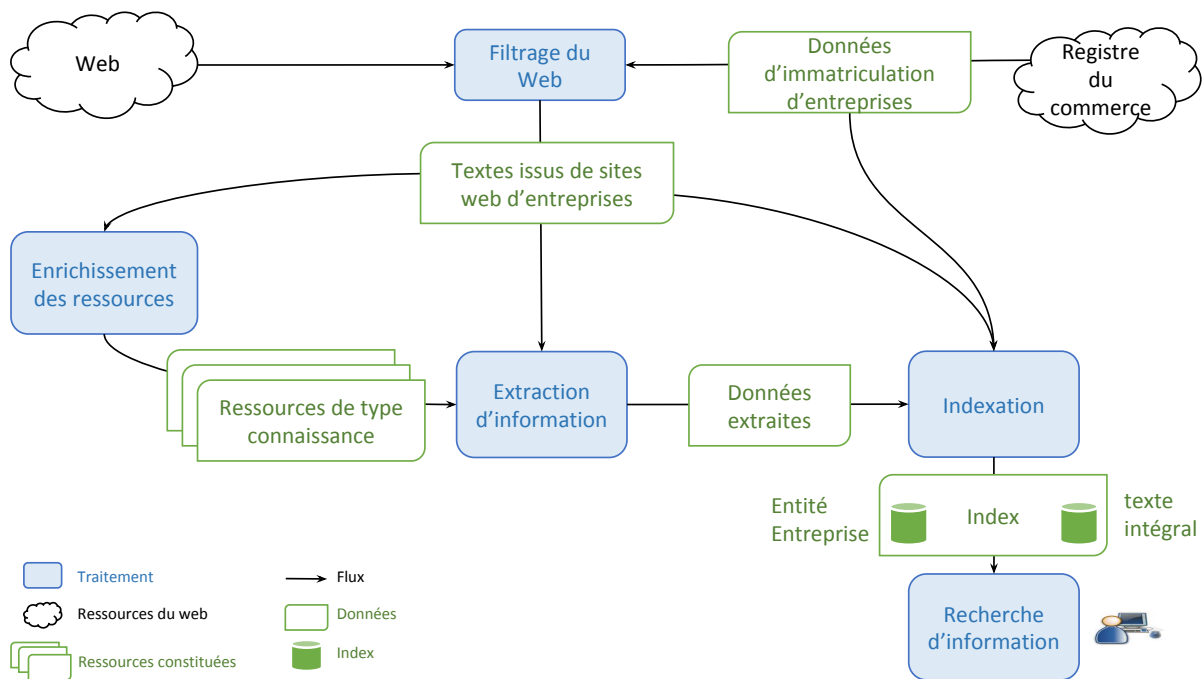


FIGURE 11.1 – Chaîne de traitement d'EN *entreprise*

des EN *entreprise*. Un troisième module (indexation) transforme, normalise et structure les informations extraites du web pour enrichir celles d'immatriculation et construire des EN *entreprise* qui sont indexées. Un quatrième module (recherche d'information) s'appuie sur les index d'EN pour mettre en œuvre des stratégies de recherche d'information. Nous avons conçu un cinquième module (enrichissement des ressources), qui exploite le contenu des sites web d'entreprises pour l'enrichissement lexical des ressources utilisées par le module d'extraction.

11.1.1 Filtrage du web

L'objectif de ce module est de filtrer le web pour identifier les sites web d'entreprises. À cet effet, nous utilisons une adaptation de l'approche par « *focused crawling* » (cf. chapitre 5, section 4.2.1.1). Nous rappelons que, dans une telle approche, un sujet (ensemble de mots clés par exemple) et une page initiale sont donnés en entrée au système. À partir de cette page initiale, le processus identifie toutes les pages qu'elle référence en se basant sur les hyperliens. Un score de pertinence est calculé pour chaque nouvelle page et celles avec un score au delà d'un seuil fixé sont sélectionnées. Ce processus est répété sur chaque nouvelle page jusqu'à atteindre un quota ou jusqu'à ce qu'il n'y ait plus de pages à analyser.

Notre processus de filtrage est décrit par la figure 11.2. Il prend en entrée une liste d'entreprises avec leurs informations d'immatriculation. Cette liste est constituée à partir des données de la plateforme open-data¹⁰¹ alimentée et maintenue par l'administration publique française. Pour l'obtenir, nous utilisons deux critères pour interroger la plateforme : un critère spatial (le département par exemple) et un critère lié au secteur d'activité (le code APE). Le processus de « *crawling* » est lancé pour chaque entreprise de la

101. <https://opendata-infogreffe.com/page/index/>

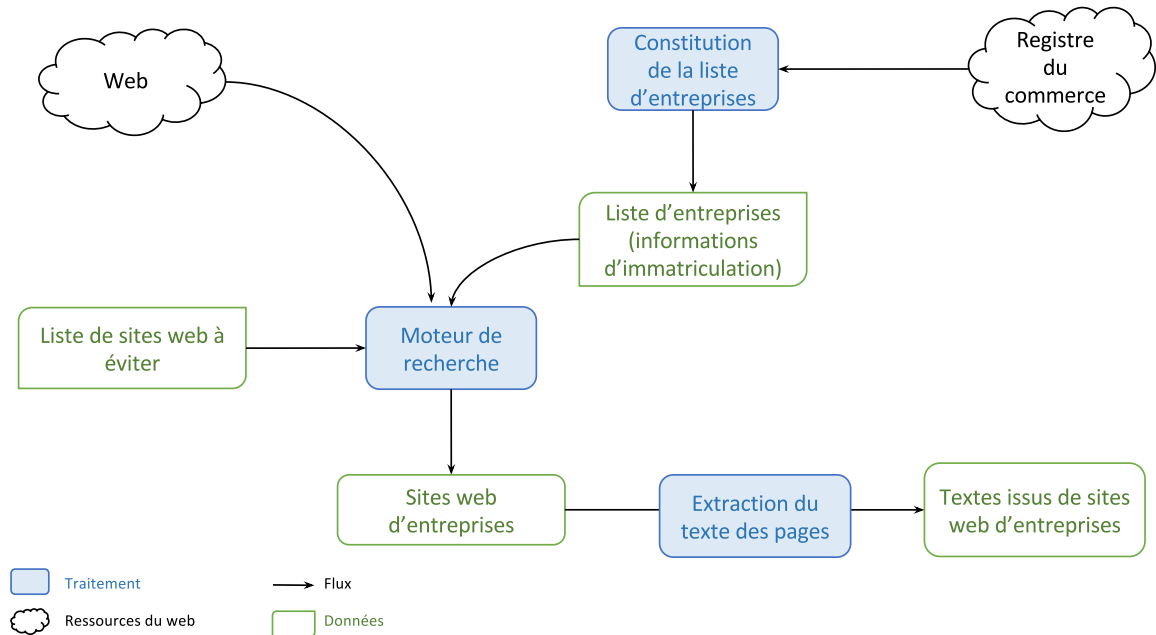


FIGURE 11.2 – Filtrage du web

liste avec comme sujet ses données d'immatriculation. La page principale d'un moteur de recherche est utilisée comme page initiale. En sortie, nous obtenons le site correspondant au résultat le plus pertinent.

L'algorithme 1 décrit en détail notre processus de filtrage du web. Une liste de domaines (sites web) à éviter est donnée en paramètre à l'algorithme. Cette liste contient un ensemble de noms de domaines web qui sont susceptibles de contenir des informations d'entreprises sans en être réellement. Elle contient, par exemple, des annuaires d'entreprises, des sites de collectivités locales, des annuaires de personnes, des sites web d'administration ou des réseaux sociaux. Pour chaque entreprise de la liste à traiter, une combinaison de ses données d'immatriculation permet d'interroger le moteur de recherche. Dans notre cas, la combinaison la plus pertinente est constituée du nom commercial de l'entreprise et du nom de la commune du siège social. Dans le cas où le nom commercial n'existe pas ou qu'aucun résultat pertinent n'a été trouvé avec ce dernier, l'utilisation du nom d'immatriculation permet aussi d'obtenir des résultats satisfaisants. En sortie de ce processus, nous avons une liste de pages. Parmi les trois pages les plus pertinentes, nous sélectionnons le premier nom de domaine qui n'est pas dans la liste des domaines à éviter.

Le texte des pages de sites web d'entreprises est extrait par la suite pour être utilisé par les modules d'enrichissement de ressources et d'extraction d'information.

Le module de filtrage est implémenté en utilisant l'outil « open-source » de création des robots d'indexation *Scrapy*¹⁰². La page initiale de notre processus de filtrage est la page de recherche du Service Google Search¹⁰³. Une fois les sites web d'entreprises identifiés, nous utilisons l'outil Apache Nutch¹⁰⁴ pour télécharger et traiter le contenu de ces pages. En sortie de ce module, nous récupérons le contenu

102. <https://scrapy.org/>

103. <http://developers.google.com/custom-search/json-api/v1/overview>

104. <http://nutch.apache.org/>

Algorithm 1 Algorithme de filtrage du web

Require: domaines à éviter

for chaque entreprise de la liste à traiter **do**
 $site_web \leftarrow Null$
 $site_web_trouve \leftarrow Faux$
 if ($nom_commercial$ existe) **then**
 $etape \leftarrow 1$
 else
 $etape \leftarrow 2$
 end if
 while ($etape < 3$) et ($non(site_web_trouve)$) **do**
 if ($etape = 1$) **then**
 $requete \leftarrow nom_commercial + nom_commune$
 else
 $requete = nom_immatriculation + nom_commune$
 end if
 interroger un moteur de recherche avec la $requete$ et récupérer les trois résultats les plus pertinents

if (tous les résultats dans la liste des domaines à éviter) **then**
 $etape \leftarrow etape + 1$
 else
 $site_web_trouve \leftarrow Vrai$
 $site_web \leftarrow$ premier site qui n'est pas dans la liste des domaines à éviter
 end if
 end while
end for

textuel des pages.

Nous avons évalué ce processus de filtrage avec un échantillon représentatif d'entreprises (jeu d'évaluation). Ce jeu d'évaluation est constitué de 600 entreprises de différents secteurs d'activités notamment, la charpente, le commerce de gros et de détail ou bien le service informatique. Pour ces 600 entreprises, l'expert a identifié un site web pour 420 d'entre elles, pendant que notre système en trouvait un pour 386 ; dont 317 étaient valables. La précision, le rappel et la F_1 -mesure correspondant à cette évaluation sont présentés dans le tableau 11.1.

Précision en %	Rappel en %	F_1 -mesure en %
82,21	77,31	79,64

TABLE 11.1 – Évaluation du module de filtrage de la chaîne *Cognisearch Business*

Notre processus de filtrage permet de retrouver correctement le site web d'environ 80% des entreprises qui en possèdent un. Les erreurs de filtrage sont généralement liées aux entreprises individuelles sans nom commercial et dont le nom d'immatriculation correspond à celui du fondateur. Pour ce cas, notre processus retourne des résultats non pertinents ne figurant pas dans la liste des domaines à ignorer, notamment des magazines et des institutions divers, faisant mention du nom ou du prénom du fondateur.

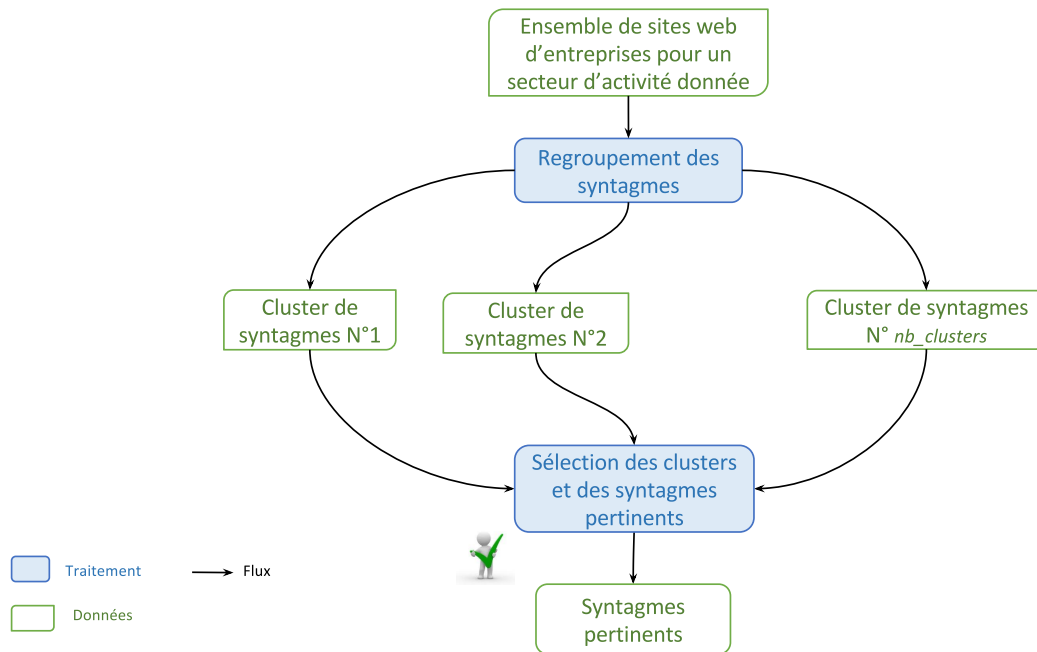


FIGURE 11.3 – Enrichissement lexical de la ressource des activités

11.1.2 Enrichissement des ressources

Nous rappelons que le module d'extraction d'information d'EN *entreprise* exploite des ressources pour extraire les données « métiers » (cf. chapitre 9). Nous avons construit ces ressources en exploitant des hiérarchies définies par l'administration publique. Par contre, le vocabulaire utilisé par celle-ci pour décrire les classes des différentes hiérarchies ne correspond que très peu à celui utilisé par les professionnels sur leurs sites web. Par conséquent, pour rendre plus efficace le processus d'extraction d'information « métiers » d'entreprises, nous mettons en œuvre un processus d'enrichissement lexical de ces ressources. Nous optons à cet effet, comme Parekh et al. [136], pour une approche par apprentissage semi-automatique basée sur le clustering. Dans la suite, nous décrivons uniquement le processus d'enrichissement de la ressource des activités, le même processus pouvant s'appliquer aux autres ressources (celles des produits et métiers).

La figure 11.3 illustre les deux phases de ce processus d'enrichissement. Pour la première phase, une technique par apprentissage non supervisé traite un ensemble de textes associées aux pages des sites web d'entreprises exerçant une même activité, et les syntagmes associés sont regroupés en clusters. Un expert intervient dans la deuxième phase pour sélectionner les clusters contenant les syntagmes pertinents pour la description de l'activité ciblée. Ces syntagmes sont ensuite ajoutés au vocabulaire de l'activité.

En effet, pour une activité donnée (concept de la ressource), un corpus de pages associées est constitué à partir des résultats obtenus via le module de filtrage. Ce corpus contient a priori des syntagmes relatifs au secteur d'activité ciblé, ainsi que ceux relatifs à d'autres vocabulaires « latents ». Deux algorithmes de clustering peuvent être utilisés pour regrouper ces syntagmes selon ces différents vocabulaires : l'algorithme LSA (Latent Semantic Allocation) [95] ou l'algorithme LDA (Latent Dirichlet Allocation) [16]. Nous avons expérimenté ces deux algorithmes dans notre contexte et les meilleurs résultats ont

été obtenus pour LDA : c'est donc ce dernier que nous utilisons pour notre implémentation. Nous traitons successivement les syntagmes correspondant aux uni-gram, aux 2-gram, aux 3-gram et aux 4-gram. L'algorithme LDA prend en entrée les paramètres suivants :

- n est la taille des syntagmes dans les clusters à constituer ;
- V correspond au vocabulaire du corpus, il s'agit de l'ensemble des syntagmes distincts dans le corpus traité ;
- M est la matrice caractérisant la relation entre le texte des pages du corpus et le vocabulaire. En effet, les colonnes de M sont constituées des éléments de V et les lignes correspondent aux textes traités. M_{ij} est le poids du j^e syntagme de V dans le texte relatif à la i^e page. Ce poids est déterminé en utilisant une adaptation du TFIDF [151] ;
- $nb_clusters$ correspond au nombre de clusters de syntagmes à construire à partir du corpus ;
- $taille_clusters$ est le nombre maximal de syntagmes par clusters.

Les syntagmes pertinents sélectionnés par l'expert au terme de ce processus, sont ajoutés à la ressource. Pour le concept relatif à l'activité « travaux de charpente », par exemple, le processus d'enrichissement lexical décrit ci-dessus a parmi de l'enrichir de 23 nouveaux labels (par exemple « démoussage de toiture », « pose d'ossature », « rénovation toiture », etc.).

Nous avons mis en œuvre le module d'enrichissement lexical des ressources en utilisant la librairie Python *lda*¹⁰⁵ qui offre une implémentation de l'algorithme LDA. Certains traitements se sont avérés nécessaires pour aligner le texte des pages web conformément aux entrées de la librairie. Par exemple, nous pouvons citer la transformation de l'ensemble des pages relatives à une catégorie d'activité en matrice (la matrice M décrite précédemment). Ces prétraitements ont aussi été réalisés avec des scripts en Python.

11.1.3 Extraction d'information

Ce module analyse le texte des pages de sites web d'entreprises pour extraire les données de contacts (téléphone, adresses, etc.) et les données « métiers » (activités, produits et services, etc.) afin de remplir la partie « données complémentaires » du modèle d'EN *entreprise* (cf. chapitre 8). Pour réaliser ce traitement, nousinstancions notre chaîne générique d'extraction d'EN complexes décrite au chapitre 9.

Ce module exploite uniquement des patrons et des ressources, nous utilisons donc la plateforme GATE¹⁰⁶ pour son implémentation. Le module OWLIM de GATE permet d'implémenter un processus d'extraction d'information en se basant sur des ressources de type connaissance au format OWL. Nous l'utilisons pour annoter les activités, métiers et produits de l'EN *entreprise*. Le module JAPE, quant à lui, est dédié à la construction des patrons pour l'extraction d'information. Nous l'exploitons pour annoter les informations de contact de l'entreprise (adresses, e-mail, numéro de téléphone, etc.). TreeTagger¹⁰⁷ a été intégré à GATE afin de pouvoir appliquer l'analyse morfo-syntaxique sur des textes en français.

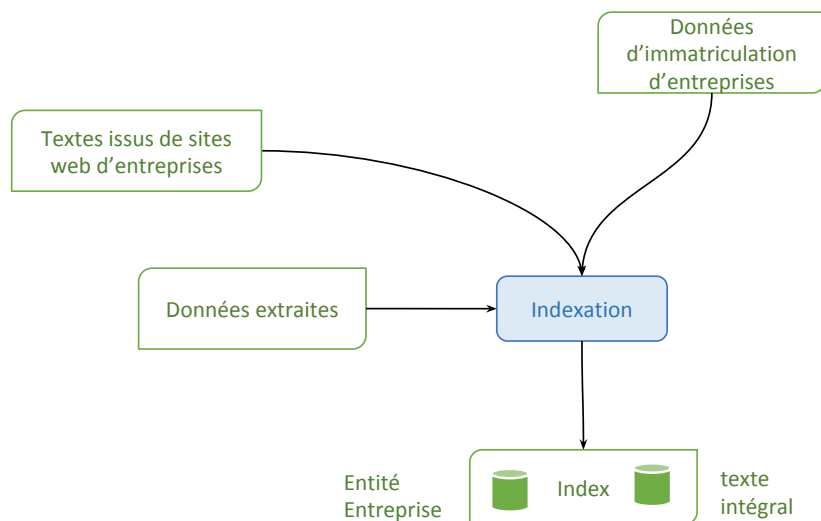
11.1.4 Indexation

La figure 11.4 décrit le processus d'indexation des EN *entreprise*. Pour chaque entreprise, les données d'immatriculation provenant de la plateforme open-data du registre du commerce et les données extraites depuis son site web sont fusionnées pour construire l'entité correspondante. Nous pondérons chaque

105. <http://pypi.python.org/pypi/lda>

106. <http://gate.ac.uk>

107. <http://www.cis.uni-muenchen.de/schmid/tools/TreeTagger/>

FIGURE 11.4 – Indexation d'EN *entreprise*

concept représentant les données « métiers » d'entreprises (activités, métiers et produits) par le nombre d'occurrences de ses labels sur le site web. Cette pondération a pour but d'ordonner les activités, métiers et produits d'une entreprise en fonction de leur importance.

Chaque adresse détectée est alignée sur le modèle d'adresse et géocodée en utilisant l'outil de l'Etalab¹⁰⁸. Les EN *entreprise* ainsi construites et normalisées sont ensuite indexées. Parallèlement, le texte intégral des pages du site web de chaque entreprise est aussi indexé.

Nous avons développé un programme JAVA qui met en œuvre tous ces traitements. Il parcourt les textes annotés par GATE pour extraire les annotations posées et les fusionne ensuite aux données d'immatriculation. Les EN et le texte des pages sont indexés sous Elasticsearch¹⁰⁹.

11.1.5 Recherche d'information

Le module de recherche d'information exploite les index alimentés par les traitements précédents pour répondre à des besoins d'information exprimés par les utilisateurs. Le processus mis en œuvre est décrit par la figure 11.5. Il supporte des critères spatiaux, thématiques et plein texte pour la recherche d'entreprises. Les mêmes ressources « métiers » et spatiales que celles exploitées en phase d'extraction d'information sont utilisées dans ce module pour interpréter la requête. Les concepts, les critères spatiaux et les mots clés de la requête sont utilisés pour interroger les index (celui des EN *entreprise* et texte intégral respectivement). En sortie de ce module, une liste d'entreprises pertinentes pour la requête exprimée est retournée.

Les requêtes sont exprimées selon la syntaxe du DSL (Domain Specific Language) propre à Elastic-

108. <https://adresse.data.gouv.fr/api/>

109. <http://www.elastic.co>

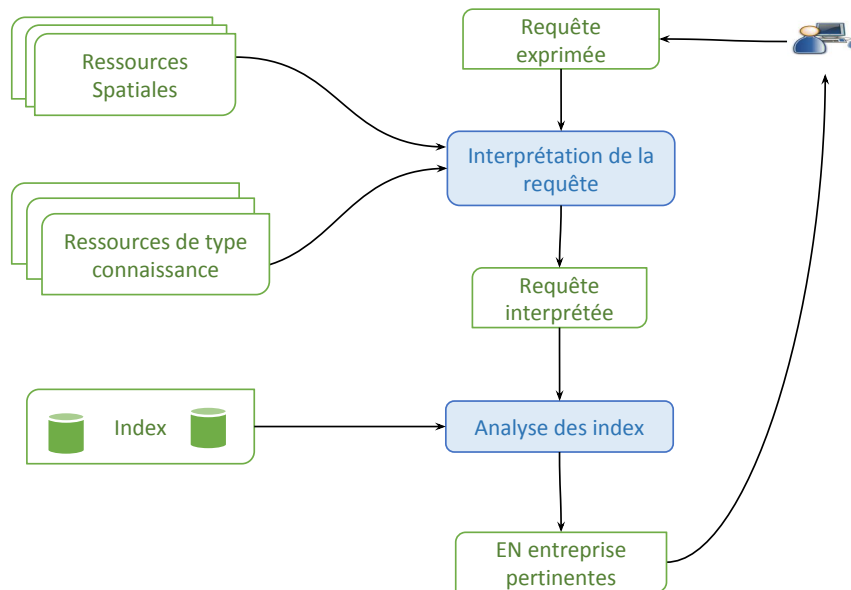


FIGURE 11.5 – Recherche d'information d'entreprises

search. L'interface de recherche s'appuie sur le plugin Marvel d'Elasticsearch¹¹⁰. L'appariement entre requête et EN indexées exploite le modèle de recherche par défaut d'Elasticsearch (TFIDF). Les résultats sont retournés sous forme d'une liste d'EN *entreprise* au format JSON et ordonnées selon leur score de pertinence.

11.1.6 Prototype du service *Cognisearch Business*

Un prototype implémentant tous les modules du service *Cognisearch Business* a été mis en œuvre. La figure 11.6 synthétise l'architecture globale du service avec pour chaque module les technologies utilisées pour son implémentation. Ce prototype s'intéresse aux entreprises de l'ancienne région Aquitaine. Nous avons identifié pour cette région un peu plus de 254 000 entreprises. Nous avons limité notre analyse à six grands secteurs d'activité correspondant à 212 codes APE. Ces secteurs d'activité ont été sélectionnés par le maître d'ouvrage (*Cogniteev*), il s'agit notamment du commerce, de la construction, de l'hébergement & la restauration, de l'enseignement, de l'information & la communication et enfin des activités scientifiques et techniques. 115 000 entreprises de notre jeu de données correspondent à ces six grands secteurs d'activité. Notre processus de filtrage du web a permis d'identifier 22 000 qui ont un site web. Pour ces 22 000 sites web, nous avons constitué un corpus de 550 000 pages web. Notons que pour chaque site, nous avons téléchargé les pages d'accueil et les pages de niveau 1 pour la constitution du corpus exploité pour cette première version de prototype. Nous désignons par pages de niveau 1, les pages directement accessibles depuis la page d'accueil d'un site web.

En ce qui concerne l'extraction d'information d'entreprises, l'analyse du corpus constitué des 550 000 pages, relatives aux 22 000 entreprises, a permis d'extraire 30 000 adresses, 44 000 syntagmes relatifs

110. <http://www.elastic.co/products/marvel>

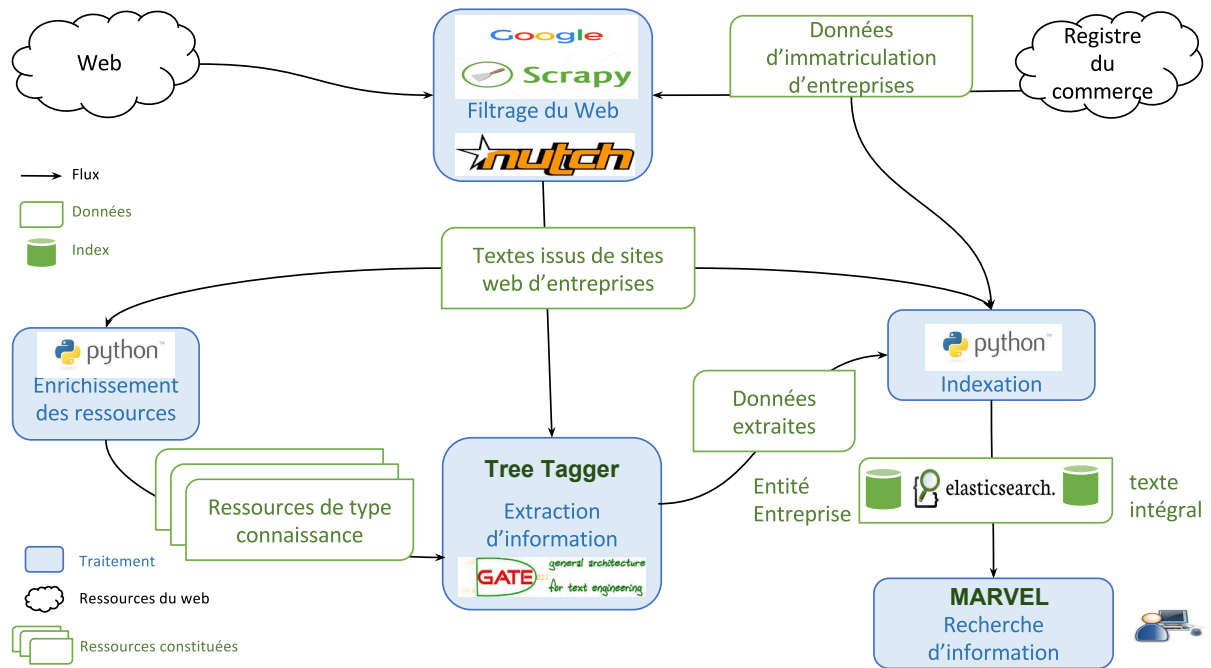


FIGURE 11.6 – Architecture technique du prototype du service Cognisearch Business

aux activités d'entreprises, 12 500 syntagmes relatifs aux produits et 28 000 autres relatifs aux métiers d'entreprises. Les deux index obtenus ont une taille totale de 3 Go.

Le téléchargement des pages des sites web, l'extraction d'information et l'indexation sont exécutés sur la plateforme Apache HADOOP¹¹¹. L'objectif étant de traiter automatiquement et efficacement un gros volume de données. L'utilisation du module *MapReduce*¹¹² d'HADOOP sur deux machines en parallèle a permis de réduire de 50% le temps des traitements.

11.2 Cognisearch Event

Le service *Cognisearch Event* est un service de recherche d'événements socio-culturels s'appuyant sur les données du web. Nous exploitons différentes sources de données pour extraire automatiquement des informations relatives à des événements. Ces informations sont structurées et normalisées selon notre modèle d'EN *événement*. Ces EN sont ensuite indexées en tenant compte du fait qu'un même *événement* peut être extrait de différentes sources. L'index constitué est interrogé pour répondre à des requêtes prenant en compte des critères spatiaux, temporels et thématiques.

L'architecture du service *Cognisearch Event* s'articule autour de quatre modules illustrés sur la figure 11.7. Le premier module est dédié aux prétraitements d'un ensemble de sites web relatifs aux événements pour ne sélectionner qu'un ensemble de pages à traiter. Le deuxième module analyse le texte de ces pages et annote des EN *événement*. Le troisième module, quant à lui, normalise les annotations et construit des EN qui sont indexées. Le dernier module est dédié à l'interrogation de l'index pour répondre

111. <http://hadoop.apache.org>

112. https://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html

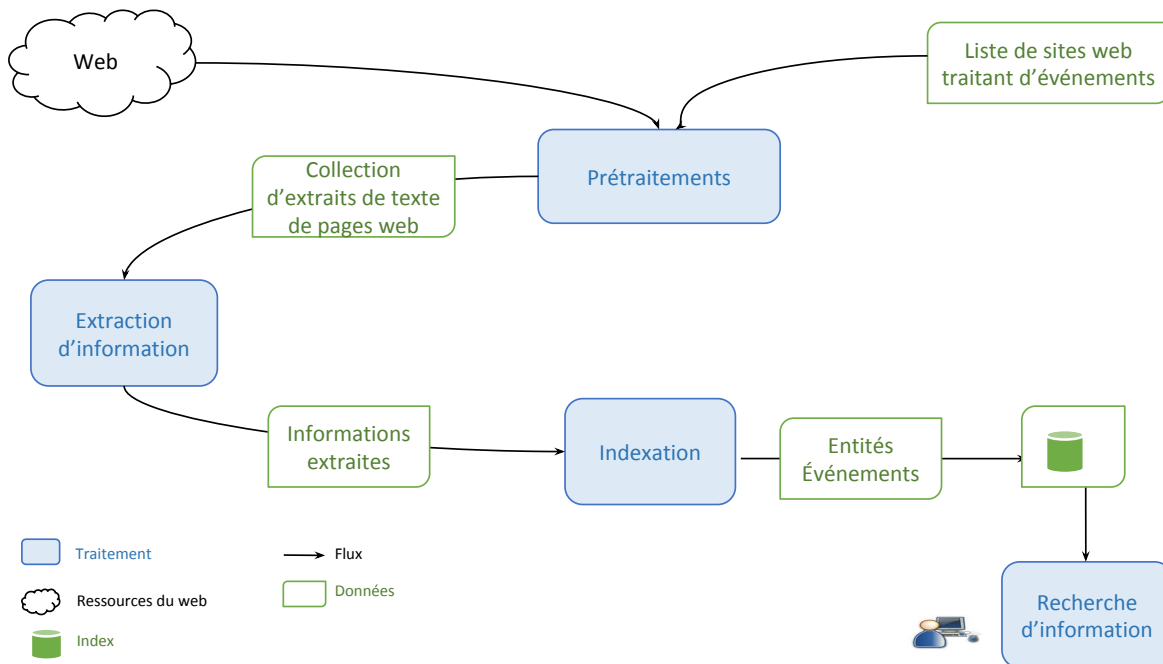


FIGURE 11.7 – Chaîne de traitement d'EN événement

à des besoins d'information.

11.2.1 Prétraitements

En entrée de ce module, nous disposons d'une liste de sites web relatifs aux événements. Les prétraitements décrits par la figure 11.8, visent : (i) le téléchargement des pages de ces sites ; (ii) les tri des pages téléchargées pour sélectionner celles susceptibles de décrire un événement ; (iii) et enfin l'extraction du texte de ces pages pour l'annotation des propriétés d'événements. Le téléchargement des pages et l'extraction du texte de ces pages se font de la même façon que pour le service *Cognisearch Business*. Nous décrivons dans la suite le tri des pages pour la sélection de celles qui seront traitées pour extraire les informations.

Le tri des pages a pour but de sélectionner dans le corpus de pages, les candidates pour l'extraction d'événements. À cet effet, nous avons conçu un classifieur binaire qui regroupe l'ensemble des pages du corpus en deux catégories : les pages « détail » et les pages « autre ».

- **Les pages « détail »** : il s'agit des pages qui correspondent à la description d'un seul *événement*. La figure 11.9 illustre un exemple de page « détail » issu du site de ticketmaster.com.
- **Les pages « autre »** : ce sont les pages du corpus qui ne sont pas des pages « détail ». Cette catégorie contient, par exemple, des pages correspondant à des brefs listings de séries d'événements avec très peu d'informations sur chacun, des pages de contact, des pages de paiement, les pages de « mentions légales » ou toute autre page ne décrivant pas un événement. La figure 11.10 illustre un exemple de page « autre » provenant du site billetréduc.com.

L'implémentation de cette phase du traitement nécessite la construction d'un classifieur binaire, qui étiquette les pages du corpus en page « détail » et page « autre ». Nous optons pour une approche par

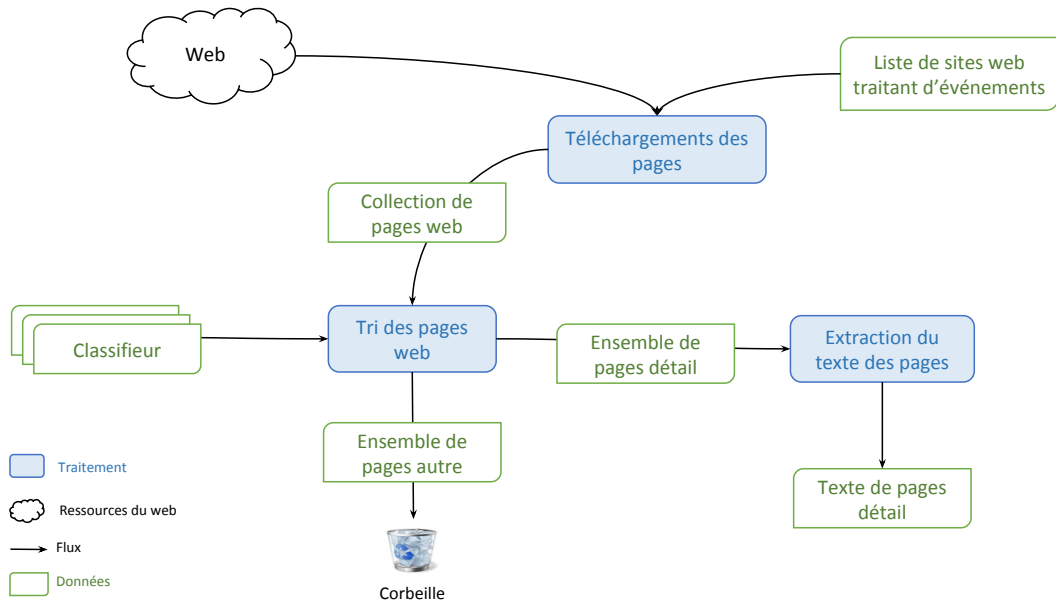


FIGURE 11.8 – Prétraitements des pages web

ticketmaster® Recherchez un spectacle, un événement, une salle ...

MUSIQUE ARTS & SPECTACLES SPORT FAMILLE & LOISIRS RÉGIONS BONNS PLANS

1- Réservation 2- Panier 3- Coordonnées 4- Paiement 5- Confirmation

KIDS UNITED ★★★★★
 CONCERT - VARIETE ET CHANSON FRANCAISE
 Artiste : KIDS UNITED

ZENITH SUD MONTPELLIER - Alerte Email Nouveautés
 Domaine de Grammont Av Albert Einstein 34000 MONTPELLIER - FRANCE

billetcollector™

BILLETTERIE
 100 % Garantie
 100 % Officielle
 e-ticket ✓
 placement ✓

Présentation	Placement et tarifs	Avis des Internauts	PMR	1 Repas Offert !
Le 21 mai, les enfants de KIDS UNITED donneront leur tout premier concert déjà complet sur la scène mythique de L'Olympia. Une consécration pour ces artistes de 8 à 15 ans dont le premier album caracole en tête des ventes depuis 2 mois. Ils seront en tournée* dans toute la France à la rentrée. * Une partie des bénéfices de la vente des billets est reversée à l'UNICEF, afin de permettre aux équipes sur le terrain de poursuivre et de renforcer leurs actions pour protéger les enfants vulnérables.				
Réservez vos places de concert pour : KIDS UNITED - ZENITH SUD MONTPELLIER Le prix des places est compris entre : 39.00 et 59.00 € Date : samedi 29 octobre 2016 Vous disposez par ailleurs du service e-ticket pour imprimer vos billets à domicile dès la fin de commande pour KIDS UNITED ainsi que du plan de salle interactif pour choisir vos places dans le lieu : ZENITH SUD MONTPELLIER.				

» Moins d'info

FIGURE 11.9 – Exemple de page « détail »



FIGURE 11.10 – Exemple de page « autre »

apprentissage supervisé. De ce fait, un échantillon représentatif de pages « détail » ou « autre » est construit à partir de l'ensemble des pages du corpus : cet échantillon constitue le jeu d'entraînement. De même, un second ensemble de pages est également constitué pour l'évaluation du classifieur : c'est le jeu de test.

Choix des critères de classification et représentation des pages - Nous utilisons la structure HTML des pages comme critères de classification. Ainsi chaque page est caractérisée par la répartition des différentes balises HTML¹¹³ qu'elle contient. Nous représentons chaque page par un vecteur des différentes balises HTML. On note n le nombre total de balises HTML définies par le W3C. Ainsi, une page web est représentée par un vecteur X de dimension n donné par :

$$X = (x_1, x_2, \dots, x_n) \in R^n, \quad x_i \text{ est le poids de la } i^e \text{ balise HTML dans le vecteur.}$$

Pondération des vecteurs représentant les pages - En ce qui concerne le calcul des x_i , nous expérimentons trois techniques afin de choisir celle qui donne les meilleurs résultats sur notre jeu de test.

— **La technique de pondération « naïve »**

Dans ce cas, le poids d'une balise dans le vecteur représentant une page correspond au nombre de ses occurrences dans celle-ci.

— **La technique de pondération normalisée**

Le poids d'une balise dans le vecteur est obtenu en normalisant le nombre d'occurrences n_i de

113. <http://www.w3schools.com/tags/>

cette balise dans la page, suivant une fonction gaussienne [160]. On note σ l'écart type entre les occurrences des différentes balises sur l'ensemble des pages du jeu d'entraînement, et μ la fréquence moyenne de chacune des balises. x_i est donné par la formule :

$$x_i = \frac{n_i - \mu}{\sigma}$$

— **La technique de pondération par changement d'échelle**

Dans ce cas, le poids de chaque balise est homogénéisé par pages, et en fonction du nombre d'occurrences des autres balises. Les valeurs obtenues des x_i oscillent entre 0 et 1. On note n_{max} le nombre d'occurrences de la balise la plus fréquente et n_{min} celui de la balise la moins fréquente dans une page web. Dans ce cas x_i est défini par :

$$x_i = \frac{n_i - n_{min}}{n_{max} - n_{min}}, \quad x_i \in [0, 1]$$

Construction du classifieur - Pour construire le classifieur, nous optons pour l'algorithme de classification SVM (Support Vector Machine) [172]. Ce choix est justifié par le fait que nous traitons d'un problème de classification binaire, pour lequel d'après Hsu et al. [82], SVM permet d'optimiser le classifieur en minimisant correctement les erreurs.

SVM s'appuie sur une méthode à base de noyaux et ceux-ci sont choisis en fonction des données traitées. Les critères de classification que nous avons choisis (balises HTML) ne sont pas forcément indépendants (imbrications de balises). Par ailleurs, le nombre de balises définis par le W3C est limité à quelques dizaines. Par conséquent, d'après les recommandations de Hsu et al. [82], le noyau radial (RBF : Radial Basic Function) est le plus pertinent dans notre contexte.

L'algorithme SVM, pour un noyau radial, prend deux paramètres : (i) $C > 0$, qui est le facteur de compromis entre les erreurs de classification et la largeur des marges entre classes ; (ii) et γ , qui est un paramètre propre au noyau radial. Nous utilisons une approche par validation croisée pour déterminer les paramètres C et γ qui optimisent le classifieur. Cette approche consiste à faire varier les paramètres dans une plage de données précise et, à chaque fois, évaluer les résultats de classification. Dans notre cas, nous faisons varier C et γ de façon exponentielle comme Hsu et al.[82].

$$C \in \{2^{-4}, 2^{-3}, \dots, 2^{11}\}$$

$$\gamma \in \{2^{-5}, 2^{-4}, \dots, 2^3, \theta\}$$

θ est l'écart type des éléments de la matrice représentant les pages du jeu d'entraînement.

Pour l'implémentation de notre classifieur, nous utilisons un jeu d'entraînement constitué de 740 pages web (une moitié de pages « détail » et l'autre de pages « autre »). Pour chaque technique de pondération, les différentes combinaisons possibles de C et γ sont utilisées pour construire les classifieurs qui sont ensuite évalués. Parmi les métriques d'évaluation de systèmes d'apprentissage listées par Mitchell et al. [121], nous avons choisi la F_1 -mesure comme métrique de comparaison. Le jeu de test de cette phase de validation est constitué de 360 autres pages web (différentes de celles du jeu d'entraînement) avec les pages

	«Naïve»	Normalisation	Changement d’échelle
Meilleur C	$2^{-1} = 0.5$	$2^3 = 8$	$2^2 = 4$
Meilleur γ	$\theta = 8.754$	$2^1 = 2$	$2^3 = 8$
F₁-Mesure	0.5202	0.9248	0.9029

TABLE 11.2 – Résultats de la validation croisée

« détail » et « autre » représentées équitablement. Le tableau 11.2 présente les résultats d’évaluation du meilleur classifieur pour chaque technique de pondération. Nous spécifions pour chacun des classifieurs, les paramètres de l’algorithme permettant de l’obtenir. L’implémentation de l’algorithme SVM que nous utilisons est celle de la librairie JAVA LIBSVM [29].

Le classifieur idéal est obtenu avec la technique de pondération par normalisation gaussienne et pour les valeurs de C et γ respectivement de 8 et de 2. Nous utilisons ce classifieur pour trier les pages du corpus et sélectionner les pages « détail ». Le texte des pages « détail » est extrait par la suite pour constituer le corpus de textes en phase d’extraction d’information.

11.2.2 Extraction d’information

Au chapitre 9, nous avons décrit notre chaîne générique dédiée à l’extraction d’EN complexes. Nous rappelons que dans son instanciation pour les EN *événement*, nous analysons le texte des pages « détail » pour annoter les informations. Certaines propriétés peuvent être bruitées (voir section 9.5) ; c’est la raison pour laquelle nous instancions un premier module pour détecter les blocs de texte contenant des propriétés d’événements. Ensuite, nous analysons ces blocs pour annoter les propriétés, et enfin, nous agrégeons les annotations posées pour construire des EN *événement*.

Pour détecter le bloc de texte correspondant au titre de l’événement, nous exploitons des patrons. Nous rappelons que l’entête d’une page HTML est contenue dans la balise `<head>...</head>`. Nous utilisons la librairie JAVA JSOUP¹¹⁴ pour analyser chaque page HTML et identifier son entête. Le texte correspondant est marqué comme un *bloc titre*.

Concernant l’implémentation des modules de détection des blocs et d’annotation de propriétés par apprentissage supervisé, nous utilisons l’algorithme CRF [116]. Le choix de CRF au détriment d’autres algorithmes d’apprentissage réputés efficaces, comme les réseaux de neurones, est justifié par la difficulté de constitution du jeu d’entraînement nécessaire à l’obtention du modèle « idéal ». En effet, les réseaux de neurones nécessitent un grand volume de données d’entraînement lorsque le nombre de caractéristiques (*features*) à prendre en compte est important [33]. Dans notre contexte, les textes analysés sont longs (600 mots en moyenne) et nous souhaitons utiliser plusieurs caractéristiques pour annoter correctement les propriétés et les blocs les contenant. CRF apparaît comme un bon compromis entre effort d’annotation manuel et processus performant.

Nous exploitons plusieurs critères pour déterminer les caractéristiques nécessaires pour l’inférence des modèles d’apprentissage. Nous nous inspirons des travaux de Ollagnier et al. [133] pour le choix de ces critères : Les plus pertinents pour notre problème sont les suivants :

- **le mot lui même** : à partir de celui-ci seront générées des caractéristiques comme la classe grammaticale (*POS : Part Of Speech*), le lemme, la forme (majuscule, minuscule, capitalisée), la position, etc. ;

114. <https://jsoup.org/>

- **la séquence de mots** : ceci permet de prendre en compte les co-occurrences de mots ;
- **les propriétés des mots d'une fenêtre** : ainsi pour chaque mot d'une fenêtre, des caractéristiques conjointes sont construites à partir de celles des autres mots. Le nombre de mots à prendre en compte à gauche et à droite du mot analysé sont des paramètres de l'algorithme ;
- **l'annotation associée aux mots d'une fenêtre** : ce critère permet de prendre en compte les annotations qui apparaissent en même temps. Notons également que ce critère est très utile pour le cas des annotations qui s'étalent sur plusieurs mots (les blocs de texte par exemple) ;
- **les sous-chaînes de caractères** : dans ce cas, les caractéristiques appliquées aux mots sont aussi appliquées aux sous-chaînes (n-gram) de ceux-ci. La taille maximale des sous-chaînes à considérer est un paramètre de l'algorithme.

Nous utilisons l'implémentation de l'algorithme CRF de la librairie *Stanford NER*¹¹⁵ pour la construction de notre système de détection de blocs et d'annotation de propriétés. Le corpus d'entraînement est constitué d'un ensemble de 360 pages « détail » extraits de 12 sites de billetterie différents. Chaque site contient 30 pages distinctes dans ce corpus. Les textes obtenus sont manuellement annotés avec l'outil *glozz*¹¹⁶ et transformés au format CoNLL 03[175].

Pour l'annotation de propriétés exploitant les ressources ou les patrons, nous utilisons la plateforme GATE pour analyser les blocs correspondants.

11.2.3 Indexation

En sortie du module d'*extraction d'information*, les annotations posées sont agrégées en EN *événement*. Ce module normalise ces annotations suivant notre modèle et compare l'EN obtenue à celles de l'index pour l'intégrer. La figure 11.11 décrit l'enchaînement des différentes étapes de l'indexation. En ce qui concerne la normalisation des EN, nous ré-utilisons l'outil de géocodage de l'Etalab pour obtenir les coordonnées géographiques des adresses de représentations d'événements. Pour les lieux définis uniquement par leur noms (« Le Bataclan », « Le Bikini », etc.), nous exploitons les ressources du web données (DBPedia et Google Maps) pour obtenir leurs coordonnées géographiques. Les noms d'intervenants sont enrichis des URI des concepts correspondant sur le web de données (DBPedia et Yago). Nous utilisons le langage SPARQL pour réaliser cet enrichissement.

Nous avons constaté en analysant les EN extraites que 73% des parties *événement* et 56% des parties *représentation* possédaient des propriétés non renseignées. Étant donné que les expérimentations des différentes approches de calcul de similarité entre EN complexes (voir chapitre 10, section 10.3.3) montrent que la régression logistique est adaptée pour ces cas de figure, nous la ré-utilisons ici pour l'intégration des EN. À partir d'une nouvelle EN extraite, le processus d'intégration mis en œuvre est le suivant :

- nous commençons par comparer la partie *événement* des EN de l'index avec celle de la nouvelle EN. Si la similarité n'est établie avec aucune des EN indexées, la nouvelle EN est intégrée directement ;
- si une EN de l'index similaire à la nouvelle est identifiée, nous comparons leurs représentations deux à deux. Les représentations de la nouvelle EN qui sont similaires à celles de l'EN existante sont fusionnées. Dans le cas contraire, nous enrichissons l'EN existante des nouvelles représentations. De même, nous fusionnons les informations des parties *événement* de la nouvelle EN avec celles de l'EN de l'index.

115. <https://nlp.stanford.edu/software/CRF-NER.shtml>

116. <http://www.glozz.org/>

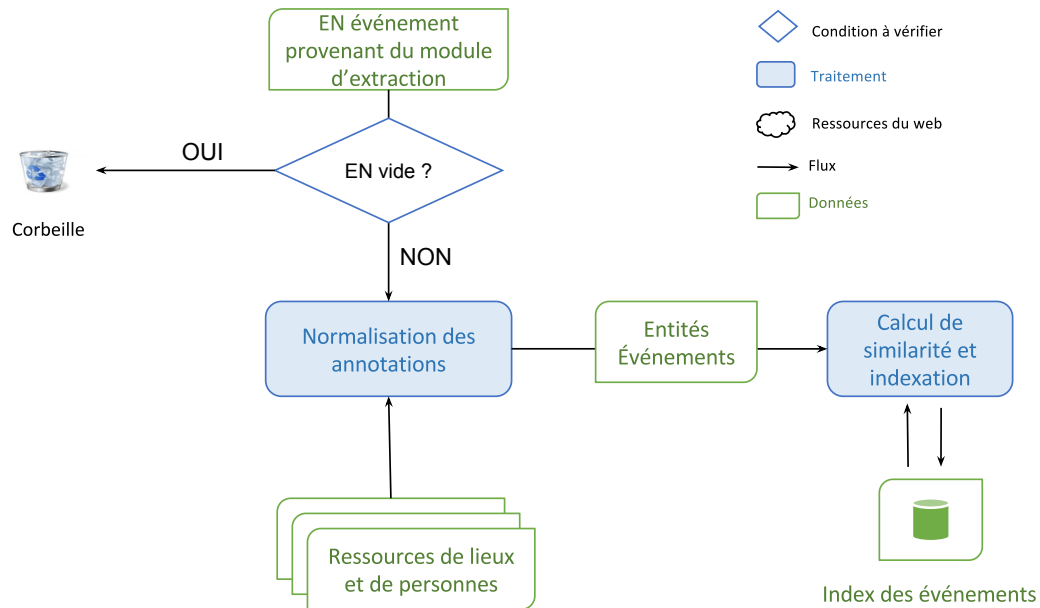


FIGURE 11.11 – Indexation d'EN *événement*

11.2.4 Recherche d'information

Pour ce module, l'utilisateur spécifie son besoin d'information et celui-ci est transformé en EN-requête. La recherche des EN pertinentes se fait en deux temps comme en phase d'indexation : (i) on commence par évaluer la similarité entre parties *événements* pour trouver les EN candidates; (ii) ensuite, nous comparons les représentations de celles-ci pour trouver celles qui correspondent à la requête traitée.

À ce stade de notre implémentation, le besoin d'information est exprimé via une interface web illustrée par la figure 11.12. Le contenu des champs renseignés est transformé en EN-requête. Cette EN-requête peut avoir des propriétés manquantes, comme l'illustre l'exemple de la figure 11.12 : le titre et l'artiste ne sont pas spécifiés pour la recherche. Pour définir sa zone de recherche, l'utilisateur spécifie un point et un rayon à prendre en considération autour de ce dernier. La recherche se fera dans le cercle défini par le point et le rayon. Le lieu peut aussi être spécifié sous forme textuelle en entrant un nom de ville ou une adresse.

En réponse à la requête, notre module retourne une liste d'EN *événement* pour lesquels on a filtré uniquement les représentations correspondant à la requête (figure 11.12). L'appariement exploite le modèle de recherche par défaut d'Elasticsearch (TFIDF) pour trouver les EN candidates. Seule la similarité entre représentations d'événements s'appuie sur une de nos approches : la régression logistique. Ce choix est justifié par le fait que l'EN-requête possède habituellement des propriétés non renseignées.

Critères de recherche

Lieu

Paris

Titre

Artiste

Catégorie

Comédie Musicale


Date

19/09/2017

Rayon (km)

5

Rechercher



The map shows the Paris region with a red pin on Paris. The map includes labels for various locations like Soissons, Evreux, Dreux, Versailles, Chartres, and Paris. It also shows major roads like A1, A4, A5, A10, A13, A15, A16, and A26. The Seine river is visible. The map is a Google Map with a search bar at the top left and navigation controls on the right.

FIGURE 11.12 – Interface de recherche d'événements

Résultat :

Titre : GREASE

Catégories :

- SPECTACLE
- COMÉDIE MUSICALE

Représentation du 19 sept. 2017

- + **Adresse :** théâtre Magador, 25 Rue de Mogador 75009 Paris
- + **Heure :** 20 h 00
- + **Artistes :** Sandy Dumbrowski [en savoir plus](#)



The map shows a detailed view of Paris, France, with a red pin on the 9th arrondissement. The map includes labels for various locations like Sarenne-Colombes, Puteaux, Neuilly-sur-Seine, Arc de Triomphe, Tour Eiffel, Grenelle, Le Jardin du Luxembourg, and Paris. It also shows major roads like D1, D7, D9, D98, D908, D909, D911, D111, N13, N185, D103, A13, and A1. The Seine river is visible. The map is a Google Map with a search bar at the top left and navigation controls on the right.

FIGURE 11.13 – Exemple de résultat pour une requête

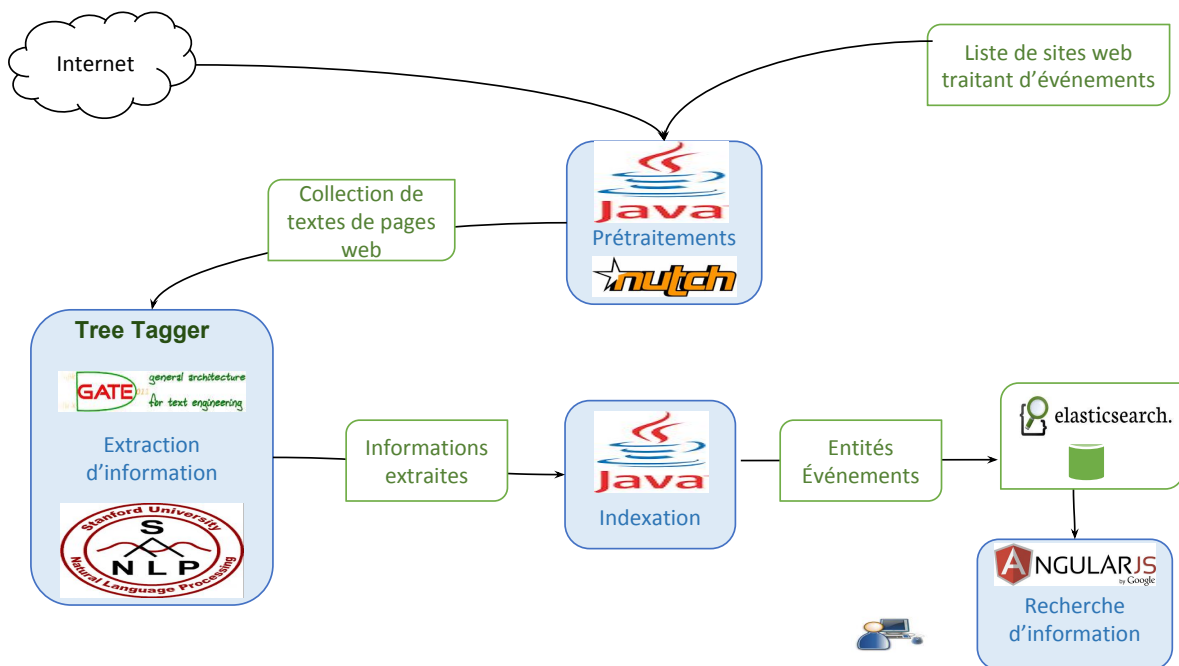


FIGURE 11.14 – Architecture technique du prototype du service *Cognisearch Event*

11.2.5 Prototype du service *Cognisearch Event*

La figure 11.14 présente l'implémentation de l'architecture du service *Cognisearch Event* avec un récapitulatif des technologies utilisées. Pour cette implémentation, nous avons traité 12 sites de billetterie en ligne, par exemple, ticketmaster.com, infoconcert.com ou encore billetréduc.com. À partir du contenu de ces sites, nous avons constitué un corpus de 23 000 pages web contenant des pages « détail » et des pages « autre ». Le classifieur a permis de sélectionner 8 250 pages « détail » de ce corpus dont nous extrayons le texte pour l'annotation d'événements.

À partir des 8 250 textes correspondant à des pages « détail », la chaîne générique d'extraction d'EN a identifié 8 100 EN *événements*. Ce qui signifie que des EN n'ont pas pu être détectées dans 150 textes analysés. En effet, dans un peu plus de la moitié de ces cas, le processus d'extraction est interrompu par des erreurs de traitement (erreurs d'analyse morpho-syntaxique principalement) ; ce qui provoque l'échec du processus d'extraction d'information. Dans d'autres cas, aucun *événement* n'est extrait des pages parce que le contenu de la page n'est plus disponible ou la page traitée n'est pas de type « détail » (erreurs de classification).

Le recherche d'événements similaires a permis de réduire le nombre d'EN *événements* à indexer à 6 320 après fusion des doublons. L'index obtenu a une taille totale de 75 Mo. Comme pour le service *Cognisearch Business*, le téléchargement des pages, l'extraction d'information et l'indexation sont mis en œuvre à travers la plateforme Apache HADOOP. L'objectif étant le même : traiter efficacement le volume de données provenant du web pour s'adapter au passage à l'échelle.

11.3 Conclusion

Nous avons présenté dans ce chapitre la mise en œuvre de notre architecture pour implémenter les services *Cognisearch Business* et *Cognisearch Event*. Les modules d'extraction d'information et ceux exploitant le calcul de similarité entre EN complexes (indexation et recherche d'information) s'appuient sur nos deux propositions génériques. En effet, nous les instancions pour traiter deux types d'EN *entreprise* et *événement*. Nous mobilisons plusieurs technologies et plateformes pour cette mise en œuvre. Les résultats d'évaluation des prototypes développés sont encourageants. Ceux-ci démontrent la pertinence de nos propositions et illustrent la faisabilité de notre architecture générale. Les deux prototypes obtenus ont été conçus de façon à s'adapter au passage à l'échelle. En d'autres termes, l'augmentation des corpus à traiter n'aura pas d'influence sur l'implémentation proposée, il suffira juste d'avoir une infrastructure adaptée et la mise en œuvre restera inchangée.

Nous envisageons le développement d'une deuxième version des services qui viendrait améliorer celle actuelle. Pour le service *Cognisearch Business*, nous allons exploiter l'apprentissage supervisé, pour améliorer la précision de notre processus d'identification de sites web d'entreprises. Concernant l'enrichissement des ressources « métiers », nous projetons d'augmenter le corpus traité pour ajouter des labels aux concepts. Concernant le service *Cognisearch Event*, nous projetons également d'enrichir la ressource des catégories en analysant plusieurs autres plateformes traitant d'événements. Par ailleurs, nous avons vu au chapitre 10 que lorsque toutes les propriétés sont renseignées, c'est l'approche par clustering combinée au modèle vectoriel qui donne de meilleurs résultats, et dans le cas contraire, c'est plutôt l'approche par régression logistique. Dans la version actuelle des prototypes, le calcul de la similarité mis en œuvre pour intégrer sans doublons les EN extraites n'exploite que la régression logistique. Dans la seconde version de nos prototypes, nous projetons plutôt de mettre en œuvre l'approche la plus adaptée selon que les EN comparées ont des propriétés non renseignées ou pas. Enfin, d'un point de vue recherche d'information, nous envisageons d'intégrer nos approches de calcul de similarité dans Elasticsearch pour apparier les besoins d'information et les EN de l'index.

Quatrième partie

Conclusion générale

Chapitre 12

Bilan : une architecture et des services génériques de recherche d'entités nommées complexes

Cadre de l'étude

La problématique traitée dans le cadre de la thèse est née d'une collaboration avec l'entreprise *Cogniteev*. Ce travail porte sur la conception de services dédiés, d'une part, à l'extraction d'EN complexes notamment *entreprise* et *événement* sur le web, et d'autre part, à la recherche d'information ciblant ces EN. L'entreprise *Cogniteev* souhaite que les approches mises en oeuvre pour le développement des services soient génériques et adaptables pour le traitement d'autres types d'EN complexes. Nous avons de ce fait conçu une architecture générale (voir figure 7.1) composée de quatre modules : (i) le premier module est dédié au traitement du contenu du web pour cibler les pages à analyser ; (ii) le deuxième module scrute le texte de ces pages pour extraire des EN complexes ; (iii) le troisième module intègre les EN extraites dans des index ; (iv) et le dernier module prend en entrée les besoins d'information des utilisateurs et parcourt les index pour rechercher les EN pertinentes. Dans cette architecture générale, nous nous sommes intéressés à l'extraction d'EN dans le texte des pages web et au calcul de similarité pour l'intégration sans doublon d'EN dans l'index et l'appariement de besoin d'information avec les EN indexées.

Contributions

Extraction d'EN complexes

Concernant l'extraction d'EN complexes, nous avons identifié plusieurs verrous pour notre contexte. Il s'agit notamment de l'extraction d'information dans un contexte bruité, de l'identification d'EN ne respectant pas de formes régulières, de l'identification d'EN dont les syntagmes sont sur-composés ou noyés dans le texte.

Nous avons conçu une chaîne générique dédiée à l'extraction d'EN complexes dans un texte. Cette

chaîne est indépendante du type d'EN traité et tient compte des différents verrous propres à notre contexte. Concernant l'extraction d'EN dans un contexte bruité, nous proposons un traitement préalable qui permet de détecter les blocs de texte contenant les propriétés ciblées. Des annotateurs dédiés sont ensuite invoqués sur ces blocs pour extraire lesdites propriétés. L'expérimentation de la chaîne d'extraction pour le cas des événements selon deux scénarios (avec et sans détection des blocs), montre un gain moyen de 40% en précision obtenu grâce à la détection de blocs.

Concernant les verrous correspondant à l'extraction d'EN suivant des formes non régulières et l'annotation d'EN sur-composées ou noyées dans le texte, nous adaptons les approches de l'état de l'art. Pour le cas des EN suivant des formes non régulières, nous exploitons des ressources pour les extraire lorsqu'elles sont exprimées dans un vocabulaire contrôlé. Dans le cas contraire, nous mettons en œuvre des techniques d'apprentissage supervisé.

Nous instancions cette chaîne générique pour extraire des EN *adresse*, *entreprise* et *événement*. Nous avons commencé par définir des modèles adéquats pour les représenter. En ce qui concerne l'annotation des adresses, la difficulté se situe au niveau de l'identification du complément d'adresse et du nom de la voie, sachant que leur position dans l'adresse peut varier et que nous ne disposons pas de ressources libres les décrivant. Pour les identifier, nous proposons deux annotateurs basés sur les patrons. Concernant l'extraction d'EN *entreprise*, la difficulté se situe au niveau de l'annotation des propriétés « métiers », qui ne suivent pas de formes régulières et sont noyées dans le texte. Pour ce problème, nous exploitons des ressources que nous avons constituées et enrichies. L'extraction des EN *événement*, quant à elle, se fait dans un contexte bruité. En effet, le texte des pages « détail » fait parfois référence à d'autres EN. Nous proposons et mettons en œuvre un traitement permettant de localiser les blocs délimitant des propriétés bruitées.

Calcul de similarité entre EN complexes

Une EN complexe est composée de plusieurs propriétés. De ce fait, pour évaluer la similarité entre deux EN complexes, il faut tout d'abord comparer deux à deux la valeur de chacune des propriétés, puis agréger le résultat de ces comparaisons. Le principal verrou dans notre contexte se situe au niveau de l'agrégation des scores de similarité entre propriétés. En effet, la même propriété peut être exprimée avec des degrés de précisions différents dans les deux EN comparées (par exemple, le lieu peut être représenté par un point dans l'une des EN et par un polygone dans l'autre). De plus, il est fréquent que certaines propriétés ne soient pas renseignées dans les EN comparées. L'enjeu est de réussir à évaluer correctement la similarité en tenant compte de ces deux facteurs.

Concernant la comparaison des propriétés, nous ré-utilisons les mesures de l'état de l'art, sauf pour les propriétés de type *lieu*, l'un est représenté par un point et l'autre par un polygone. Nous proposons une nouvelle formule de calcul qui prend en compte la périphérie du polygone dans le calcul de sa similarité avec un point.

Concernant l'agrégation des scores de similarité entre propriétés, nous proposons trois approches :

- La première approche, nommée « **approche par étalonnage** », reprend la combinaison linéaire pour agréger les scores de similarité entre propriétés. Contrairement à la méthode classique pour laquelle les poids de ces différentes similarités sont déterminés de façon empirique, nous mettons en œuvre une technique par étalonnage.
- La deuxième approche, nommée « **approche par régression logistique** », évalue la similarité

globale comme la probabilité que les EN comparées soient similaires, sachant les différents scores de similarité entre propriétés. Nous exploitons la technique d’apprentissage basée sur la régression logistique pour calculer cette probabilité.

- La troisième approche, nommée « **approche basée sur les arbres** », exploite la technique d’aide à la décision basée sur les arbres de décision pour agréger les différents scores de similarité entre propriétés.

Nous proposons également une nouvelle approche de calcul de similarité entre EN complexes, sans calculer les scores de similarité entre propriétés. Cette approche, nommé « **approche par clustering** », combine de façon originale les techniques de clustering au modèle vectoriel pour comparer des EN complexes. En effet, le clustering est utilisé pour partitionner l’ensemble d’EN complexes en clusters correspondant aux différentes propriétés. Ensuite, nous représentons ces EN comme des vecteurs de scores d’appartenance à l’ensemble de clusters. La similarité entre deux EN est obtenue en évaluant le cosinus de l’angle entre les vecteurs les représentant.

Nous avons expérimenté ces différentes approches dédiées au calcul de similarité entre EN complexes. Nous avons constaté que l’approche par clustering permet d’obtenir les meilleurs résultats sur nos jeux d’évaluation lorsque les EN sont totalement renseignées. Dans le cas où au moins une des EN contient des propriétés manquantes, les approches par apprentissage (régression logistique et arbres de décision) sont les plus adaptées. En effet, ces approches exploitent l’information relative au fait que certaines EN puissent être partiellement renseignées pour inférer le score de similarité au niveau des EN complexes.

Prototypes

Nous avons mis en œuvre l’architecture générale que nous avons conçue, pour démontrer sa faisabilité et expérimenter nos différentes contributions. Cette architecture a été implémentée pour le développement d’une première version des deux services demandés par *Cogniteev* : *Cognisearch Business* et *Cognisearch Event*. Le service *Cognisearch Business* est construit à partir du traitement des sites web de 22 000 entreprises. De même, nous analysons 12 plateformes de billetterie pour la mise en œuvre du service *Cognisearch Event*. De ces 12 plateformes, nous avons extraits un peu plus de 8 000 événements. Nous mobilisons un ensemble d’outils et de plateformes pour l’implémentation des différents modules de chaque service, notamment GATE, HADOOP ou encore la librairie Stanford NLP.

Perspectives

Extraction d’information sur le web

Dans nos travaux, nous analysons le contenu textuel des pages pour extraire les informations. Les balises HTML sont peu exploitées dans le processus mis sur pied. Nous les exploitons, par exemple, pour l’identification des blocs de texte contenant le titre des événements. Or l’un des enjeux même de la structuration des pages suivant le langage HTML est de mettre en avant et/ou de regrouper les informations de celles-ci. En effet, une information contenue dans une balise `<h1> ... </h1>`, n’a pas la même importance que celle contenue dans une balise ` ... `. De même, la balise `<div> ... </div>` est généralement utilisée pour regrouper les informations dans une page web. Par ailleurs, la proportion de sites sur le web exploitant les métadonnées a connu une forte croissance depuis le début de la thèse.

À ce jour par exemple, plus de 10 millions de sites web utilisent les métadonnées de type *schema.org*¹¹⁷ pour structurer sémantiquement leurs contenus. Parmi les 21 plateformes traitant d'événements que nous avons observées au début de la thèse, 2 seulement exploitaient les métadonnées. À ce jour, nous en dénombrons déjà 5. Au regard de ces deux observations, nous projetons d'exploiter d'avantage le balisage HTML, ainsi que les métadonnées pour renforcer notre chaîne générique d'extraction d'EN complexes, et plus précisément la détection de blocs basée sur l'apprentissage. Le modèle d'apprentissage serait désormais construit en exploitant les balises HTML et les métadonnées, en plus du texte des pages (propriétés linguistiques et morphologiques des mots).

Concernant l'extraction d'information basée sur les ressources de type connaissance, nous avons vu que les mots du texte et ceux de la ressource subissent un ensemble de traitements (tokenisation, lemmatisation, etc.) avant la projection de la ressource sur le texte. Considérons, par exemple, que le texte contienne le syntagme « *charpente en bois traditionnelle* » et que la ressource contienne plutôt « *charpente traditionnelle en bois* ». Ces deux syntagmes sont tous les deux constitués des mêmes lemmes, mais ceux-ci ne sont pas dans le même ordre. Par conséquent, le texte ne sera pas annoté. Il serait intéressant d'exploiter des mesures de calcul de similarité entre textes courts, qui ne tiennent pas compte de l'ordre des mots pour réaliser la projection de la ressource sur le texte. L'enjeu serait d'évaluer le gain en rappel obtenu par rapport à l'impact sur la précision.

Calcul de similarité

Nous avons proposé des approches de calcul de similarité entre EN complexes, que nous avons évaluées d'un point de vue qualitatif. En effet, nous avons calculé la précision, le rappel, la F_1 -mesure et l'exactitude sur des jeux d'évaluation constitués de centaines de couples d'EN. Une perspective serait d'évaluer également nos approches en termes de temps de traitement ou de ressources nécessaires pour ce calcul. En fait, l'approche combinant le clustering au modèle vectoriel qui est la plus performante pour comparer deux EN ayant toutes les propriétés renseignées, s'appuie sur la comparaison de chaque propriété avec tous les clusters correspondants pour construire les vecteurs d'EN. De ce fait, si l'ensemble d'EN analysé est grand (plusieurs milliers), il y a de fortes chances d'avoir un grand nombre de clusters et donc plusieurs comparaisons à faire pour la construction des vecteurs d'EN. Ce qui peut nécessiter beaucoup de mémoire et de temps de traitement. De même, pour l'approche par étalonnage qui donne de bons rappels, lorsque les EN complexes ont un grand nombre de propriétés, la combinatoire pour générer les n-uplets à expérimenter peut devenir « exponentielle ».

Une dernière perspective cible le module de recherche d'information de notre architecture générale (cf. 7.1). En effet, nous avons développé des prototypes qui offrent des interfaces pour l'interrogation des informations extraites du web. Dans ces prototypes, nous représentons le besoin d'information comme une EN complexe (EN-requête) dans laquelle les propriétés correspondent aux différents critères de recherche, ceux-ci pouvant parfois être non renseignés. Le processus d'appariement consiste à déterminer les EN de l'index similaires à l'EN-requête, ceci en exploitant nos approches de calcul de similarité. La perspective pour la recherche d'information s'articule autour de deux objectifs :

- le premier vise l'évaluation de nos approches de calcul de similarité dans un contexte de RI. Pour cet objectif, nous observerons deux particularités : (i) le comportement de nos approches lorsque le nombre de critères non renseignés est élevé ; (ii) le temps moyen de traitement nécessaire pour

117. <http://schema.org/>

- appairer l'EN-requête avec les EN de l'index pour chacune des approches ;
- le second objectif est d'évaluer l'adaptabilité des nos approches de calcul de similarité lorsqu'il s'agit de prendre en compte les préférences des utilisateurs. En effet, nous envisageons de permettre aux utilisateurs, lors de la spécification de leurs requêtes, d'associer des priorités aux différents critères. Nous souhaitons ainsi adapter nos différentes approches de calcul de similarité en les dotant de degrés de préférence, de neutralité ou de rejet.

Bibliographie

- [1] Alan AGRESTI et Maria KATERI. *Categorical data analysis*. Springer, 2011.
- [2] Dirk AHLERS. “Business entity retrieval and data provision for yellow pages by local search”. In : *IRPS Workshop@ ECIR2013*. 2013.
- [3] Dirk AHLERS et Susanne BOLL. “Retrieving Address-based Locations from the Web”. In : *Proceedings of the 2Nd International Workshop on Geographic Information Retrieval*. GIR '08. Napa Valley, California, USA : ACM, 2008, p. 27–34. ISBN : 978-1-60558-253-5. DOI : 10.1145/1460007.1460015. URL : <http://doi.acm.org/10.1145/1460007.1460015>.
- [4] Massimiliano ALBANESE et VS SUBRAHMANIAN. “T-REX : A domain-independent system for automated cultural information extraction”. In : *First International Conference on Computational Cultural Dynamics (ICCCD 2007)*. 2007.
- [5] John H ALDRICH et Forrest D NELSON. *Linear probability, logit, and probit models*. T. 45. Sage, 1984.
- [6] Omar ALONSO, Ricardo BAEZA-YATES, Jannik STRÖTGEN et Michael GERTZ. “Temporal Information Retrieval : Challenges and Opportunities”. In : *In : 1st Temporal Web Analytics Workshop at WWW*. 2011, p. 1–8.
- [7] Sylvain ARLOT, Alain CELISSE et al. “A survey of cross-validation procedures for model selection”. In : *Statistics surveys* 4. 2010, p. 40–79.
- [8] Ricardo BAEZA-YATES, Berthier RIBEIRO-NETO et al. *Modern information retrieval*. T. 463. ACM press New York, 1999.
- [9] Mohammad BAHRAM, Urmas KOLJALG, Pierre-Emmanuel COURTY, Abdala G DIEDHIOU, Rasmus KJØLLER, Sergei POLME, Martin RYBERG, Vilmar VELDRE et Leho TEDERSOO. “The distance decay of similarity in communities of ectomycorrhizal fungi in different ecosystems and scales”. In : *Journal of Ecology* 101.5. Wiley Online Library, 2013, p. 1335–1344.
- [10] Maurice S BARTLETT. “A note on the multiplying factors for various χ^2 approximations”. In : *Journal of the Royal Statistical Society. Series B (Methodological)*. JSTOR, 1954, p. 296–298.
- [11] Kate BEARD et Vyjayanti SHARMA. “Multidimensional ranking for data in digital spatial libraries”. In : *International Journal on Digital Libraries* 1.2. Springer, 1997, p. 153–160.
- [12] Hila BECKER, Mor NAAMAN et Luis GRAVANO. “Learning similarity metrics for event identification in social media”. In : *Proceedings of the third ACM international conference on Web search and data mining*. ACM. 2010, p. 291–300.

- [13] Imene BENSALÉM et Mohamed-Kiredine KHOLLADI. “La désambiguïsation des toponymes par la densité géographique”. In : *International Conference on Applied Informatics (ICAI'09)*. 2009.
- [14] Daniel M BIKEL, Scott MILLER, Richard SCHWARTZ et Ralph WEISCHEDEL. “Nymble : a high-performance learning name-finder”. In : *Proceedings of the fifth conference on Applied natural language processing*. Association for Computational Linguistics. 1997, p. 194–201.
- [15] Frédéric BILHAUT et Antoine WIDLÖCHER. “LinguaStream : An Integrated Environment for Computational Linguistics Experimentation”. In : *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics : Posters38 ; Demonstrations*. EACL '06. Trento, Italy : Association for Computational Linguistics, 2006, p. 95–98.
- [16] David M BLEI, Andrew Y NG et Michael I JORDAN. “Latent dirichlet allocation”. In : *the Journal of machine Learning research* 3. JMLR. org, 2003, p. 993–1022.
- [17] Sebastian BLOHM. *Large-scale pattern-based information extraction from the world wide web*. KIT Scientific Publishing, 2011.
- [18] Olivier BODENREIDER. “The unified medical language system (UMLS) : integrating biomedical terminology”. In : *Nucleic acids research* 32.suppl 1. Oxford Univ Press, 2004, p. D267–D270.
- [19] Karla A. V. BORGES, Alberto H. F. LAENDER, Claudia B. MEDEIROS et Clodoveu A. DAVIS Jr. “Discovering Geographic Locations in Web Pages Using Urban Addresses”. In : *Proceedings of the 4th ACM Workshop on Geographical Information Retrieval*. GIR '07. Lisbon, Portugal : ACM, 2007, p. 31–36. ISBN : 978-1-59593-828-2. DOI : 10.1145/1316948.1316957. URL : <http://doi.acm.org/10.1145/1316948.1316957>.
- [20] Karla A. BORGES, Clodoveu A. DAVIS Jr, Alberto H. LAENDER et Claudia Bauzer MEDEIROS. “Ontology-driven Discovery of Geospatial Evidence in Web Pages”. In : *Geoinformatica* 15.4. Hingham, MA, USA : Kluwer Academic Publishers, oct. 2011, p. 609–631. ISSN : 1384-6175. DOI : 10.1007/s10707-010-0118-z. URL : <http://dx.doi.org/10.1007/s10707-010-0118-z>.
- [21] Jethro BORSJE, Frederik HOGENBOOM et Flavius FRASINCAR. “Semi-automatic financial events discovery based on lexico-semantic patterns.” In : *Int. J. Web Eng. Technol.* 6.2. 2010, p. 115–140.
- [22] Andrew BORTHWICK. “A maximum entropy approach to named entity recognition”. Thèse de doct. Citeseer, 1999.
- [23] Henrik BULSKOV, Rasmus KNAPPE et Troels ANDREASEN. “On measuring similarity for conceptual querying”. In : *International Conference on Flexible Query Answering Systems*. Springer. 2002, p. 100–111.
- [24] Davide BUSCALDI et Bernardo MAGNINI. “Grounding toponyms in an Italian local news corpus”. In : *Proceedings of the 6th Workshop on Geographic Information Retrieval*. ACM. 2010, p. 15.
- [25] Deng CAI, Shipeng YU, Ji-Rong WEN et Wei-Ying MA. “Extracting Content Structure for Web Pages Based on Visual Representation”. In : *5th Asia-Pacific Web Conference on Web Technologies and Applications*. APWeb'03. Xian, China : Springer-Verlag, 2003, p. 406–417. ISBN : 3-540-02354-2. URL : <http://dl.acm.org/citation.cfm?id=1766091.1766143>.
- [26] Claire CARDIE. “Empirical methods in information extraction”. In : *AI magazine* 18.4. 1997, p. 65.

-
- [27] Peggy CELLIER, Thierry CHARNOIS, Marc PLANTEVIT, Christophe RIGOTTI, Bruno CRÉMILLEUX, Olivier GANDRILLON, Jiří KLÉMA et Jean-Luc MANGUIN. “Sequential pattern mining for discovering gene interactions and their contextual information from biomedical texts”. In : *Journal of biomedical semantics* 6.1. BioMed Central, 2015, p. 27.
- [28] Chia-Hui CHANG, Mohammed KAYED, Moheb Ramzy GIRGIS et Khaled F. SHAALAN. “A Survey of Web Information Extraction Systems”. In : *IEEE Trans. on Knowl. and Data Eng.* 18.10. Piscataway, NJ, USA : IEEE Educational Activities Department, 2006, p. 1411–1428. ISSN : 1041-4347. DOI : 10.1109/TKDE.2006.152. URL : <http://dx.doi.org/10.1109/TKDE.2006.152>.
- [29] Chih-Chung CHANG et Chih-Jen LIN. “LIBSVM : A library for support vector machines”. In : *ACM Transactions on Intelligent Systems and Technology* 2. 2011, 27 :1–27 :27. URL : <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [30] Ben CHOI et Zhongmei YAO. “Web Page Classification.” In : *Studies in Fuzziness and Soft Computing*. T. 180. Springer Berlin / Heidelberg, 2015, p. 221–274. DOI : 10.1007/11362197_9.
- [31] Nitin R CHOPDE et M NICHAT. “Landmark based shortest path detection by using a* and haversine formula”. In : *International Journal of Innovative Research in Computer and Communication Engineering* 1.2. 2013, p. 298–302.
- [32] William COHEN, Pradeep RAVIKUMAR et Stephen FIENBERG. “A comparison of string metrics for matching names and records”. In : *Kdd workshop on data cleaning and object consolidation*. T. 3. 2003, p. 73–78.
- [33] Ronan COLLOBERT et Jason WESTON. “A Unified Architecture for Natural Language Processing : Deep Neural Networks with Multitask Learning”. In : *Proceedings of the 25th International Conference on Machine Learning*. ICML '08. Helsinki, Finland : ACM, 2008, p. 160–167. ISBN : 978-1-60558-205-4. DOI : 10.1145/1390156.1390177. URL : <http://doi.acm.org/10.1145/1390156.1390177>.
- [34] A. CORRADI, G. CURATOLA, L. FOSCHINI, R. IANNIELLO et C. R. De ROLT. “Automatic extraction of POIs in smart cities : Big data processing in ParticipAct”. In : *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. Mai 2015, p. 1059–1064. DOI : 10.1109/INM.2015.7140433.
- [35] Célia da COSTA PEREIRA, Mauro DRAGONI et Gabriella PASI. “Multidimensional relevance : Prioritized aggregation in a personalized Information Retrieval setting”. In : *Information processing & management* 48.2. Elsevier, 2012, p. 340–357.
- [36] Silviu CUCERZAN. “Large-scale named entity disambiguation based on Wikipedia data”. In : 2007.
- [37] Hamish CUNNINGHAM, Diana MAYNARD et Kalina BONTCHEVA. *Text processing with GATE*. Gateway Press CA, 2011.
- [38] Pierre DAGNELIE. *Statistique théorique et appliquée*. T. 2. De Boeck Université, 1998.
- [39] Laurence DANLOS. “ILIMP : Outil pour repérer les occurrences du pronom impersonnel il”. In : *Actes de TALN* 5. 2005, p. 123–132.

- [40] Jesse DAVIS et Mark GOADRICH. “The Relationship Between Precision-Recall and ROC Curves”. In : *Proceedings of the 23rd International Conference on Machine Learning*. ICML '06. Pittsburgh, Pennsylvania, USA : ACM, 2006, p. 233–240. ISBN : 1-59593-383-2. DOI : 10.1145/1143844.1143874. URL : <http://doi.acm.org/10.1145/1143844.1143874>.
- [41] Arindam DEY, Abhijit PAUL et Bipul Syam PURKAYASTHA. “Named Entity Recognition for Nepali language : A Semi Hybrid Approach”. In : *International Journal of Engineering and Innovative Technology (IJEIT) Volume 3*. 2014, p. 21–25.
- [42] Michel Marie DEZA et Elena DEZA. “Encyclopedia of distances”. In : *Encyclopedia of Distances*. Springer, 2009, p. 1–583.
- [43] Michelangelo DILIGENTI, Frans COETZEE, Steve LAWRENCE, C. Lee GILES et Marco GORI. “Focused Crawling Using Context Graphs”. In : *Proceedings of the 26th International Conference on Very Large Data Bases*. VLDB '00. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 2000, p. 527–534. ISBN : 1-55860-715-3. URL : <http://dl.acm.org/citation.cfm?id=645926.671854>.
- [44] Li DING, Tim FININ, Anupam JOSHI, Rong PAN, R. Scott COST, Yun PENG, Pavan REDDIVARI, Vishal DOSHI et Joel SACHS. “Swoogle : A Search and Metadata Engine for the Semantic Web”. In : *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*. CIKM '04. Washington, D.C., USA : ACM, 2004, p. 652–659. ISBN : 1-58113-874-1. DOI : 10.1145/1031171.1031289. URL : <http://doi.acm.org/10.1145/1031171.1031289>.
- [45] Yoann DUPONT. “La structuration dans les entités nommées”. Thèse de doct. Paris 3, 2017.
- [46] Russell EBERHART et James KENNEDY. “A new optimizer using particle swarm theory”. In : *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on*. IEEE. 1995, p. 39–43.
- [47] Maud EHRMANN. “Les entités nommées, de la linguistique au TAL : statut théorique et méthodes de désambiguïsation”. Thèse de doct. Paris 7, 2008.
- [48] S Anitha ELAVARASI, J AKILANDESWARI et K MENAGA. “A survey on semantic similarity measure”. In : *International Journal of Research in Advent Technology* 2.3. 2014, p. 389–398.
- [49] Oren ETZIONI, Michael CAFARELLA, Doug DOWNEY, Ana-Maria POPESCU, Tal SHAKED, Stephen SODERLAND, Daniel S WELD et Alexander YATES. “Unsupervised named-entity extraction from the web : An experimental study”. In : *Artificial intelligence* 165.1. Elsevier, 2005, p. 91–134.
- [50] Peter EXNER et Pierre NUGUES. “Using semantic role labeling to extract events from Wikipedia”. In : *Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2011). Workshop in conjunction with the 10th International Semantic Web Conference*. 2011, p. 23–24.
- [51] Nicolas FAESSEL. “Indexation et interrogation de pages Web décomposées en blocs visuels”. Thèse de doct. Aix Marseille 3, 2011.
- [52] Herbert FEDERER. *Geometric measure theory*. Springer, 2014.
- [53] Alfio FERRARAM, Andriy NIKOLOV et François SCHARFFE. “Data linking for the semantic web”. In : *Semantic Web : Ontology and Knowledge Base Enabled Tools, Services, and Applications* 169. Idea Group Inc., 2013, p. 326.

-
- [54] Michele FILANNINO, Gavin BROWN et Goran NENADIC. “ManTIME : Temporal expression identification and normalization in the TempEval-3 challenge”. In : *arXiv preprint arXiv :1304.7942*. 2013.
- [55] John FOLEY, Michael BENDERSKY et Vanja JOSIFOVSKI. “Learning to Extract Local Events from the Web”. In : *38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '15. Santiago, Chile : ACM, 2015, p. 423–432. ISBN : 978-1-4503-3621-5. DOI : 10.1145/2766462.2767739. URL : <http://doi.acm.org/10.1145/2766462.2767739>.
- [56] Armel T. FOTSOH, Annig LE PARC - LACAYRELLE et Tanguy MOAL. “Cognisearch Business : un service de recherche d’entreprises sur le web”. In : *Proceedings of GAST 2016, Atelier Gestion et Analyse des données Spatiales et Temporelles*. GAST'16. Reims, France, 2016.
- [57] Armel T. FOTSOH, Christian SALLABERRY et Annig LE PARC - LACAYRELLE. “Calcul de similarité entre événements sociaux”. In : *Proceedings of EXCES 2016, Atelier EXtraction de Connaissances à partir de données Spatialisées*. EXCES'17. Rouen, France, 2017.
- [58] Armel T. FOTSOH, Christian SALLABERRY, Annig LE PARC - LACAYRELLE et Tanguy MOAL. “A process for business entities extraction on the web”. In : *Proceedings of iSWAG 2016, The Second International Symposium on Web Algorithms*. iSWAG '16. Deauville, France, 2016.
- [59] Armel T. FOTSOH, Christian SALLABERRY, Annig LE PARC - LACAYRELLE et Tanguy MOAL. “Extraction of Business Information on the web to Supply a Geolocated Search Service”. In : *Proceedings of ALLDATA 2016, The Second International Conference on Big Data, Small Data, Linked Data and Open Data*. AllData '16. Lisbon, Portugal : IARIA, 2016, p. 82–85. ISBN : 978-1-61208-457-2. DOI : 10.1145/1316948.1316957.
- [60] Armel FOTSOH, Christian SALLABERRY et Annig LE PARC - LACAYRELLE. “Named entity similarity computation”. In : *GIR'17 : 11th Workshop on Geographic Information Retrieval, November 30-December 1, 2017*. GIR'17. ACM. Heidelberg, Germany, 2017.
- [61] Nordine FOUROUR. “Nemesis, un système de reconnaissance incrémentielle des entités nommées pour le français”. In : *Actes de la 9ème conférence sur le Traitement Automatique des Langues Naturelles (TALN'02)*. 2002, p. 265–274.
- [62] Edward A FOX et Joseph A SHAW. “Combination of multiple searches”. In : *NIST SPECIAL PUBLICATION SP 243*. NATIONAL INSTITUTE OF STANDARDS & TECHNOLOGY, 1994.
- [63] J FRIEDMAN et Bogdan E POPESCU. *Gradient directed regularization for linear regression and classification*. Rapp. tech. Citeseer, 2003.
- [64] Mauro GAIO et Ludovic MONCLA. “Extended named entity recognition using finite-state transducers : An application to place names”. In : *The Ninth International Conference on Advanced Geographic Information Systems, Applications, and Services (GEOProcessing 2017)*. 2017.
- [65] Raul GARCIA-CASTRO. *Benchmarking semantic web technology*. T. 3. IOS Press, 2009.
- [66] Roger GODEMENT. *Cours d’algèbre*. T. 5. Hermann Paris, 1966.
- [67] Christos GOUMOPOULOS, Eleni CHRISTOPOULOU, Nikos DROSSOS et Achilles KAMEAS. “The PLANTS system : enabling mixed societies of communicating plants and artefacts”. In : *European Symposium on Ambient Intelligence*. Springer. 2004, p. 184–195.
- [68] Gregory GREFENSTETTE. “Comparing two language identification schemes”. In : 1995.

- [69] Ralph GRISHMAN et Beth SUNDHEIM. “Message Understanding Conference-6 : A Brief History.” In : *Coling*. T. 96. 1996, p. 466–471.
- [70] Thomas R GRUBER et al. “A translation approach to portable ontology specifications”. In : *Knowledge acquisition* 5.2. Academic Press, 1993, p. 199–220.
- [71] Fabrice GUILLET, Gilbert RITSCHARD, Djamel Abdelkader ZIGHED, Henri BRIAND et al. *Advances in Knowledge Discovery and Management*. T. 2. Springer, 2012.
- [72] Chetan GUPTA et Robert GROSSMAN. “Genic : A single pass generalized incremental algorithm for clustering”. In : *Proceedings of the 2004 SIAM International Conference on Data Mining*. SIAM. 2004, p. 147–153.
- [73] Dhruv GUPTA. “Event Search and Analytics : Detecting Events in Semantically Annotated Corpora for Search & Analytics”. In : *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*. WSDM '16. San Francisco, California, USA : ACM, 2016, p. 705–705. ISBN : 978-1-4503-3716-8. DOI : 10.1145/2835776.2855083. URL : <http://doi.acm.org/10.1145/2835776.2855083>.
- [74] Willem Robert van HAGE, Véronique MALAISE, Roxane H SEGERS, Laura HOLLINK et Guus SCHREIBER. “Design and use of the Simple Event Model (SEM)”. In : *Web Semantics : Science, Services and Agents on the World Wide Web* 9.2. Elsevier, 2011. ISSN : 1570-8268. URL : <http://www.websemanticsjournal.org/index.php/ps/article/view/190>.
- [75] Maria HALKIDI, Benjamin NGUYEN, Iraklis VARLAMIS et Michalis VAZIRGIANNIS. “THESUS : Organizing Web document collections based on link semantics”. In : *The VLDB Journal—The International Journal on Very Large Data Bases* 12.4. Springer-Verlag New York, Inc., 2003, p. 320–332.
- [76] Arezki HAMMACHE et Rachid AHMED-OUAMER. “Système d’Inférence pour une Indexation de Documents Basée sur une Ontologie de Domaine.” In : *INFORSID*. 2006, p. 895–910.
- [77] Robin HANSON, John STUTZ et Peter CHEESEMAN. “Bayesian classification theory”. In : 1991.
- [78] Farah HARRATHI, Catherine ROUSSEY, Loïc MAISONNASSE et Sylvie CALABRETTO. “Vers une approche statistique pour l’indexation sémantique des documents multilingues”. In : *28ème congrès INFORSID*. 2010, p–127.
- [79] Peter HARRINGTON. *Machine learning in action*. T. 5. Manning Greenwich, CT, 2012.
- [80] J. T. HASTINGS. “Automated Conflation of Digital Gazetteer Data”. In : *Int. J. Geogr. Inf. Sci.* 22.10. Bristol, PA, USA : Taylor & Francis, Inc., jan. 2008, p. 1109–1127. ISSN : 1365-8816.
- [81] Linda Ladd HILL. “Access to geographic concepts in online bibliographic files : effectiveness of current practices and the potential of a graphic interface”. In : University Microfilms International (UMI), 1990.
- [82] Chih-Wei HSU, Chih-Chung CHANG, Chih-Jen LIN et al. “A practical guide to support vector classification”. In : 2003.
- [83] P JACCARD. “Bulletin de la Société Vaudoise des Sciences Naturelles”. In : *Etude comparative de la distribution florale dans une portion des Alpes et des Jura* 37. 1901, p. 547–579.
- [84] Krzysztof JANOWICZ, Martin RAUBAL et Werner KUHN. “The semantics of similarity in geographic information retrieval”. In : *Journal of Spatial Information Science* 2011.2. 2012, p. 29–57.

- [85] Matthew A JARO. “Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida”. In : *Journal of the American Statistical Association* 84.406. Taylor & Francis, 1989, p. 414–420.
- [86] Ridong JIANG, Rafael E BANCHS et Haizhou LI. “Evaluating and combining named entity recognition systems”. In : *Proceedings of the Sixth Named Entity Workshop, Joint with 54th Association for Computational Linguistics*. 2016, p. 21–27.
- [87] Taghva K., Coombs J., Pereda R. et Nartker T. “Address extraction using hidden markov models”. In : 2005, p. 119–126.
- [88] P. KALVA, F. ENEMBRECK et A. KOERICH. “WEB Image Classification Based on the Fusion of Image and Text Classifiers”. In : *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*. T. 1. Sept. 2007, p. 561–568. DOI : 10.1109/ICDAR.2007.4378772.
- [89] Houda KHROUF et Raphael TRONCY. “De la modélisation sémantique des événements vers l’enrichissement et la recommandation.” In : *Revue d’Intelligence Artificielle* 28.2-3. 2014, p. 321–347.
- [90] Houda KHROUF et Raphaël TRONCY. “EventMedia : A LOD dataset of events illustrated with media”. In : *Semantic Web 7.2*. IOS Press, 2016, p. 193–199.
- [91] Junchul KIM, Maria VASARDANI et Stephan WINTER. “Similarity matching for integrating spatial information extracted from place descriptions”. In : *International Journal of Geographical Information Science* 31.1. Taylor & Francis, 2017, p. 56–80.
- [92] Christian KOHLSCHÜTTER, Peter FANKHAUSER et Wolfgang NEJDL. “Boilerplate Detection Using Shallow Text Features”. In : *Proceedings of the Third ACM International Conference on Web Search and Data Mining*. WSDM ’10. New York, New York, USA : ACM, 2010, p. 441–450. ISBN : 978-1-60558-889-6. DOI : 10.1145/1718487.1718542. URL : <http://doi.acm.org/10.1145/1718487.1718542>.
- [93] Hans-Peter KRIEGEL et Matthias SCHUBERT. “Classification of Websites as Sets of Feature Vectors.” In : *Databases and applications*. 2004, p. 127–132.
- [94] Erdal KUZHEY et Gerhard WEIKUM. “Extraction of Temporal Facts and Events from Wikipedia”. In : *Proceedings of the 2Nd Temporal Web Analytics Workshop*. TempWeb ’12. Lyon, France : ACM, 2012, p. 25–32. ISBN : 978-1-4503-1188-5. DOI : 10.1145/2169095.2169101. URL : <http://doi.acm.org/10.1145/2169095.2169101>.
- [95] T.K. LANDAUER, P.W. FOLTZ et D. LAHAM. “An introduction to latent semantic analysis”. In : *Discourse processes* 25. ALEX PUBLISHING CO, 1998, p. 259–284.
- [96] Thomas LARGILLIER, Guillaume PEYRONNET et Sylvain PEYRONNET. “Efficient filtering of adult content using textual information”. In : *CoRR* abs/1512.00198. 2015. URL : <http://arxiv.org/abs/1512.00198>.
- [97] C LE MEUR, S GALLIANO et E GEOFFROIS. *Conventions d’annotations en Entités Nommées*. 2004.
- [98] Annig LE PARC-LACAYRELLE, Mauro GAIO et Christian SALLABERRY. “La composante temps dans l’information géographique textuelle.” In : *Document Numérique* 10.2. Lavoisier, 2007, p. 129–148. URL : <https://hal.archives-ouvertes.fr/hal-00389639>.

BIBLIOGRAPHIE

- [99] Ngoc Tan LE, Fatma MALLEK et Fatiha SADAT. “UQAM-NTL : Named entity recognition in Twitter messages”. In : *WNUT 2016*. 2016, p. 197.
- [100] Joon Ho LEE. “Analyses of multiple evidence combination”. In : *ACM SIGIR Forum*. T. 31. SI. ACM. 1997, p. 267–276.
- [101] Gaël LEJEUNE, Romain BRIXTTEL, Antoine DOUCET et Nadine LUCAS. “Multilingual event extraction for epidemic detection”. In : *Artificial intelligence in medicine* 65.2. Elsevier, 2015, p. 131–143.
- [102] Cane Wing ki LEUNG, Jing JIANG, Kian Ming Adam CHAI, Hai Leong CHIEU et Loo-Nin TEOW. “Unsupervised Information Extraction with Distributional Prior Knowledge.” In : *EMNLP. ACL*, 2011, p. 814–824. ISBN : 978-1-937284-11-4.
- [103] Vladimir I LEVENSHTAIN. “Binary codes capable of correcting deletions, insertions, and reversals”. In : *Soviet physics doklady*. T. 10. 8. 1966, p. 707–710.
- [104] Hao LI, Heng JI et Lin ZHAO. “Social Event Extraction : Task, Challenges and Techniques”. In : *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*. ASONAM '15. Paris, France : ACM, 2015, p. 526–532. ISBN : 978-1-4503-3854-7. DOI : 10.1145/2808797.2809413. URL : <http://doi.acm.org/10.1145/2808797.2809413>.
- [105] Hui LI, Jannik STRÖTGEN, Julian ZELL et Michael GERTZ. “Chinese Temporal Tagging with HeidelTime.” In : *EACL*. T. 2014. 2014, p. 133–7.
- [106] Wenwen LI, Michael F. GOODCHILD, Richard L. CHURCH et Bin ZHOU. “Geospatial Data Mining on the Web : Discovering Locations of Emergency Service Facilities”. In : 2012, p. 552–563. DOI : 10.1007/978-3-642-35527-1_46. URL : http://dx.doi.org/10.1007/978-3-642-35527-1_46.
- [107] Anne-Laure LIGOZAT, Brigitte GRAU, Isabelle ROBBA et Anne VILNAT. “L’extraction des réponses dans un système de question-réponse”. In : *Traitement Automatique des Langues Naturelles (TALN 2006)*. 2006.
- [108] Dekang LIN et al. “An information-theoretic definition of similarity.” In : *ICML*. T. 98. 1998. Citeseer. 1998, p. 296–304.
- [109] Shuyang LIN, Fengjiao WANG, Qingbo HU et Philip S YU. “Extracting social events for learning better information diffusion models”. In : *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2013, p. 365–373.
- [110] Bing LIU, Robert GROSSMAN et Yanhong ZHAI. “Mining Data Records in Web Pages”. In : *Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '03. Washington, D.C. : ACM, 2003, p. 601–606. ISBN : 1-58113-737-0. DOI : 10.1145/956750.956826. URL : <http://doi.acm.org/10.1145/956750.956826>.
- [111] Berenike LOOS et Chris BIEMANN. “supporting web-based address extraction with unsupervised tagging”. In : *Data Analysis, Machine Learning and Applications 2008*. 2008, p. 577–584.
- [112] Thusitha MABOTUWANA, Michael C LEE et Eric V COHEN-SOLAL. “An ontology-based similarity measure for biomedical data—Application to radiology reports”. In : *Journal of biomedical informatics* 46.5. Elsevier, 2013, p. 857–868.

-
- [113] Christopher D. MANNING, Prabhakar RAGHAVAN et Hinrich SCHÜTZE. *Introduction to Information Retrieval*. New York, NY, USA : Cambridge University Press, 2008.
- [114] Stefania MARRARA, Gabriella PASI et Marco VIVIANI. “Aggregation operators in Information Retrieval”. In : *Fuzzy Sets and Systems* 324. 2017, p. 3–19. DOI : 10.1016/j.fss.2016.12.018. URL : <https://doi.org/10.1016/j.fss.2016.12.018>.
- [115] B. MARTINS, H. MANGUINHAS et J. BORBINHA. “Extracting and Exploring the Geo-Temporal Semantics of Textual Resources”. In : *Proceedings of the International Conference on Semantic Computing*. IEEE Computer Society, août 2008, p. 1–9. DOI : 10.1109/ICSC.2008.86. URL : http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4597167.
- [116] Andrew McCALLUM et Wei LI. “Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons”. In : *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*. Association for Computational Linguistics. 2003, p. 188–191.
- [117] Grant MCKENZIE, Krzysztof JANOWICZ et Benjamin ADAMS. “A weighted multi-attribute method for matching user-generated points of interest”. In : *Cartography and Geographic Information Science* 41.2. Taylor & Francis, 2014, p. 125–137.
- [118] Sara MESHKIZADEH et Amir Masoud RAHMANI. “Webpage Classification based on Compound of Using HTML Features & URL Features and Features of Sibling Pages”. In : *Int. J. Adv. Comp. Techn.* 2. 2010, p. 36–46.
- [119] George A. MILLER. “WordNet : A Lexical Database for English”. In : *Commun. ACM* 38.11. New York, NY, USA : ACM, nov. 1995, p. 39–41. ISSN : 0001-0782. DOI : 10.1145/219717.219748. URL : <http://doi.acm.org/10.1145/219717.219748>.
- [120] Teruko MITAMURA. *TAC KBP event detection annotation guidelines*. Rapp. tech. v1. 7. Technical report, Carnegie Mellon University, September, 2014.
- [121] Tom M MITCHELL et al. *Machine learning*. 1997.
- [122] Ludovic MONCLA et Mauro GAIO. “A multi-layer markup language for geospatial semantic annotations”. In : *Proceedings of the 9th Workshop on Geographic Information Retrieval*. ACM. 2015, p. 5.
- [123] Alvaro E MONGE, Charles ELKAN et al. “The Field Matching Problem : Algorithms and Applications.” In : *KDD*. 1996, p. 267–270.
- [124] Erwan MOREAU, François YVON et Olivier CAPPÉ. “Robust similarity measures for named entities matching”. In : *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*. Association for Computational Linguistics. 2008, p. 593–600.
- [125] Antonio MUCHERINO, Petraq J PAPAJORGJI et Panos M PARDALOS. “K-nearest neighbor classification”. In : *Data Mining in Agriculture*. Springer, 2009, p. 83–106.
- [126] David NADEAU et Satoshi SEKINE. “A survey of named entity recognition and classification”. In : *Linguisticae Investigationes* 30.1. John Benjamins publishing company, 2007, p. 3–26.
- [127] Robert NECHES, Richard E FIKES, Tim FININ, Thomas GRUBER, Ramesh PATIL, Ted SENATOR et William R SWARTOUT. “Enabling technology for knowledge sharing”. In : *AI magazine* 12.3. 1991, p. 36.

- [128] Paolo NESI, Gianni PANTALEO et Marco TENTI. “Geographical localization of web domains and organization addresses recognition by employing natural language processing, Pattern Matching and clustering”. In : *Engineering Applications of Artificial Intelligence* 51. Elsevier, 2016, p. 202–211.
- [129] Tien NGUYEN VAN, Mauro GAIO et Christian SALLABERRY. “Recherche de relations spatio-temporelles : une méthode basée sur l’analyse de corpus textuels”. In : *TIA ’09WS : Acquisition et modélisation de relations sémantiques*. Toulouse, France, nov. 2009, p. 1–6. URL : <https://hal.archives-ouvertes.fr/hal-00452005>.
- [130] Van Tien NGUYEN, Christian SALLABERRY et Mauro GAIO. “Mesure de la similarité entre termes et labels de concepts ontologiques”. In : *arXiv preprint arXiv :1307.6422*. 2013.
- [131] Andriy NIKOLOV, Mathieu D’AQUIN et Enrico MOTTA. “Unsupervised learning of link discovery configuration”. In : *The Semantic Web : Research and Applications*. Springer, 2012, p. 119–133.
- [132] NIST. “The ACE 2005 (ACE05) Evaluation Plan”. In : 2005.
- [133] Anaïs OLLAGNIER, Sébastien FOURNIER et Patrice BELLOT. “Cascade de CRFs et SVM pour la détection de références bibliographiques diffuses dans les articles scientifiques.” In : *CORIA-CIFED*. 2016, p. 139–152.
- [134] Salvatore ORLANDO, Francesco PIZZOLON et Gabriele TOLOMEI. “SEED : A Framework for Extracting Social Events from Press News”. In : *22Nd International Conference on World Wide Web. WWW ’13 Companion*. Rio de Janeiro, Brazil : ACM, 2013, p. 1285–1294. ISBN : 978-1-4503-2038-2. DOI : 10.1145/2487788.2488163. URL : <http://doi.acm.org/10.1145/2487788.2488163>.
- [135] Damien PALACIO. “Combinaison de critères par contraintes pour la Recherche d’Information Géographique”. Thèse de doct. Université de Pau et des Pays de l’Adour, 2010.
- [136] Viral PAREKH, Jack GWO et Tim FINIM. “Mining Domain Specific Texts and Glossaries to Evaluate and Enrich Domain Ontologies”. In : *International Conference of Information and Knowledge Engineering*. 2004. URL : http://ebiquity.umbc.edu/%5C_file%5C_directory%5C_/papers/94.pdf.
- [137] Giulia Di PIETRO, Carlo ALIPRANDI, Antonio Ercole De LUCA, Matteo RAFFAELLI et Tiziana SORU. “Semantic crawling : An approach based on Named Entity Recognition”. In : *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2014, Beijing, China, August 17-20, 2014*. 2014, p. 695–699. DOI : 10.1109/ASONAM.2014.6921661. URL : <http://dx.doi.org/10.1109/ASONAM.2014.6921661>.
- [138] James PUSTEJOVSKY, José M CASTANO, Robert INGRIA, Roser SAURI, Robert J GAIZAUSKAS, Andrea SETZER, Graham KATZ et Dragomir R RADEV. “TimeML : Robust specification of event and temporal expressions in text.” In : *New directions in question answering* 3. 2003, p. 28–34.
- [139] J. Ross QUINLAN. “Induction of decision trees”. In : *Machine learning* 1.1. Springer, 1986, p. 81–106.
- [140] Roy RADA, Hafedh MILI, Ellen BICKNELL et Maria BLETTNER. “Development and application of a metric on semantic nets”. In : *IEEE transactions on systems, man, and cybernetics* 19.1. IEEE, 1989, p. 17–30.

-
- [141] Adam RAE, Vanessa MURDOCK, Adrian POPESCU et Hugues BOUCHARD. “Mining the Web for Points of Interest”. In : *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '12. Portland, Oregon, USA : ACM, 2012, p. 711–720. ISBN : 978-1-4503-1472-5. DOI : 10.1145/2348283.2348379. URL : <http://doi.acm.org/10.1145/2348283.2348379>.
- [142] Calin RAILEAN et Alexandra MORARU. “Discovering popular events from tweets”. In : *Proceedings of the 16th International Multiconference Information Society, Ljubljana, Slovenia*. 2013.
- [143] Yves RAIMOND, Samer A. ABDALLAH, Mark SANDLER et Frederick GIASSON. “The Music Ontology”. In : *Proceedings of the 8th International Conference on Music Information Retrieval*. Vienna, Austria, sept. 2007, p. 417–422.
- [144] Ram Kumar RANA et Nidhi TYAGI. “Article : A Novel Architecture of Ontology-based Semantic Web Crawler”. In : *International Journal of Computer Applications* 44.18. Avr. 2012, p. 31–36.
- [145] M RANGEL VICENTE. “La glose comme outil de désambiguïsation référentielle des noms propres”. In : *Corela, Numéros spéciaux, Le traitement lexicographique des noms propres*. URL : <http://edel.univ-poitiers.fr/corela/document.php>. 2005.
- [146] Nils REITER et Paul BUITELAAR. “Lexical Enrichment of a Human Anatomy Ontology using WordNet”. In : *Information Retrieval in Biomedicine : Natural Language Processing for Knowledge Integration, IGI Global, 2009*. 5th Global WordNet Conference. University of Szeged, 2008, p. 375–387.
- [147] Philip RESNIK. “Using information content to evaluate semantic similarity in a taxonomy”. In : *arXiv preprint cmp-lg/9511007*. 1995.
- [148] Ellen RILOFF et Janyce WIEBE. “Learning Extraction Patterns for Subjective Expressions”. In : *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*. EMNLP '03. Stroudsburg, PA, USA : Association for Computational Linguistics, 2003, p. 105–112. DOI : 10.3115/1119355.1119369. URL : <http://dx.doi.org/10.3115/1119355.1119369>.
- [149] Alan RITTER, MAUSAM, Oren ETZIONI et Sam CLARK. “Open Domain Event Extraction from Twitter”. In : *18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '12. Beijing, China : ACM, 2012, p. 1104–1112. ISBN : 978-1-4503-1462-6. DOI : 10.1145/2339530.2339704. URL : <http://doi.acm.org/10.1145/2339530.2339704>.
- [150] Lior ROKACH. “Decision forest : Twenty years of research”. In : *Information Fusion* 27. Elsevier, 2016, p. 111–125.
- [151] François ROUSSEAU et Michalis VAZIRGIANNIS. “Graph-of-word and TW-IDF : New Approach to Ad Hoc IR”. In : *Proceedings of the 22Nd ACM International Conference on Conference on Information & Knowledge Management*. CIKM '13. San Francisco, California, USA : ACM, 2013, p. 59–68. ISBN : 978-1-4503-2263-8. DOI : 10.1145/2505515.2505671. URL : <http://doi.acm.org/10.1145/2505515.2505671>.
- [152] S.L. RUEBEN et G. JAKOBSON. *Digital maps displaying search-resulting points-of-interest in user delimited regions*. US Patent 8,510,045. 2013. URL : <https://www.google.com/patents/US8510045>.

- [153] Christian SALLABERRY, Mustapha BAZIZ, Julien LESBEGUERIES et Mauro GAIO. “Towards an IE and IR System Dealing with Spatial Information in Digital Libraries-Evaluation Case Study.” In : *ICEIS (5)*. 2007, p. 190–197.
- [154] Christian SALLABERRY, Mauro GAIO, Damien PALACIO et Julien LESBEGUERIES. “Fuzzifying GIS Topological Functions for GIR Needs”. In : *Proceedings of the 2Nd International Workshop on Geographic Information Retrieval*. GIR '08. Napa Valley, California, USA : ACM, 2008, p. 1–8. ISBN : 978-1-60558-253-5. DOI : 10.1145/1460007.1460008. URL : <http://doi.acm.org/10.1145/1460007.1460008>.
- [155] Gerard SALTON. “Introduction to modern information retrieval”. In : *McGraw-Hill*. 1983.
- [156] Gerard SALTON et Christopher BUCKLEY. “Term-weighting approaches in automatic text retrieval”. In : *Information processing & management* 24.5. Elsevier, 1988, p. 513–523.
- [157] Gerard SALTON, Anita WONG et Chung-Shu YANG. “A vector space model for automatic indexing”. In : *Communications of the ACM* 18.11. ACM, 1975, p. 613–620.
- [158] Daniel SANCHEZ-CISNEROS et Fernando Aparicio GALI. “UEM-UC3M : An Ontology-based named entity recognition system for biomedical texts.” In : *SemEval@ NAACL-HLT*. 2013, p. 622–627.
- [159] Frédéric SANTOS. “Arbres de décision”. In : 2015.
- [160] Gilbert SAPORTA. *Probabilités, analyse des données et statistique*. Editions Technip, 2006.
- [161] Tatjana SCHEFFLER, Rafael SCHIRRU et Paul LEHMANN. “Matching points of interest from different social networking sites”. In : *KI 2012 : Advances in Artificial Intelligence*. Springer, 2012, p. 245–248.
- [162] Sebastian SCHMIDT, Simon MANSCHITZ, Christoph RENSING et Ralf STEINMETZ. “Extraction of address data from unstructured text using free knowledge resources”. In : *Proceedings of the 13th International Conference on Knowledge Management and Knowledge Technologies*. ACM. 2013, p. 7.
- [163] Satoshi SEKINE et al. “NYU : Description of the Japanese NE system used for MET-2”. In : *Proc. Message Understanding Conference*. 1998.
- [164] Laurie SERRANO. “Vers une capitalisation des connaissances orientée utilisateur : Extraction et structuration automatiques de l’information issue de sources ouvertes”. Thèse de doct. Université de Caen Basse-Normandie, 2014.
- [165] Laurie SERRANO, Maroua BOUZID, Thierry CHARNOIS, Stephan BRUNESSAUX et Bruno GRILHERES. “Extraction et agrégation automatique d’événements pour la veille en sources ouvertes : du texte à la connaissance”. In : *IC-24èmes Journées francophones d’Ingénierie des Connaissances*. 2013.
- [166] Khaled SHAALAN et Hafsa RAZA. “Person Name Entity Recognition for Arabic”. In : *Proceedings of the 2007 Workshop on Computational Approaches to Semitic Languages : Common Issues and Resources*. Semitic '07. Prague, Czech Republic : Association for Computational Linguistics, 2007, p. 17–24. URL : <http://dl.acm.org/citation.cfm?id=1654576.1654581>.
- [167] Claude E. SHANNON. “A mathematical theory of communication, bell System technical Journal 27 : 379-423 and 623–656”. In : *Mathematical Reviews (MathSciNet) : MR10, 133e*. 1948.

- [168] KC SHET, U Dinesh ACHARYA et al. “A new similarity measure for taxonomy based on edge counting”. In : *arXiv preprint arXiv :1211.4709*. 2012.
- [169] Yusuke SHINYAMA et Satoshi SEKINE. “Named entity discovery using comparable news articles”. In : *Proceedings of the 20th international conference on Computational Linguistics*. Association for Computational Linguistics. 2004, p. 848.
- [170] Kara SONER, Alan OZGUR, Sabuncu ORKUNT, Akpınar SAMET, Cicekli Nihan K. et Alpaslan Ferda N. “An Ontology-based Retrieval System Using Semantic Indexing”. In : t. 37. 4. Oxford, UK, UK, juin 2012, p. 294–305. DOI : 10.1016/j.is.2011.09.004. URL : <http://dx.doi.org/10.1016/j.is.2011.09.004>.
- [171] Jannik STRÖTGEN et Michael GERTZ. “Heideltime : High quality rule-based extraction and normalization of temporal expressions”. In : *Proceedings of the 5th International Workshop on Semantic Evaluation*. Association for Computational Linguistics. 2010, p. 321–324.
- [172] Johan AK SUYKENS et Joos VANDEWALLE. “Least squares support vector machine classifiers”. In : *Neural processing letters 9.3*. Springer, 1999, p. 293–300.
- [173] Charles TEISSÈDRE, Delphine BATTISTELLI et Jean-Luc MINEL. “Du texte au portail sémantique : cas d’utilisation lié à des données temporelles”. In : *IC 2010*. Sous la dir. de Sylvie DESPRES. France : Ecole des Mines d’Alès, juin 2010, p. 209–220. URL : <https://halshs.archives-ouvertes.fr/halshs-00493727>.
- [174] W. THAMVISET et S. WONGTHANAVASU. “Bottom-up region extractor for semi-structured web pages”. In : *Computer Science and Engineering Conference (ICSEC), 2014 International*. Juil. 2014, p. 284–289. DOI : 10.1109/ICSEC.2014.6978209.
- [175] Erik F. TJONG KIM SANG et Fien DE MEULDER. “Introduction to the CoNLL-2003 Shared Task : Language-independent Named Entity Recognition”. In : *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*. CONLL ’03. Edmonton, Canada : Association for Computational Linguistics, 2003, p. 142–147. DOI : 10.3115/1119176.1119195. URL : <http://dx.doi.org/10.3115/1119176.1119195>.
- [176] Raphaël TRONCY, Ryan SHAW et Lynda HARDMAN. “LODE : une ontologie pour représenter des événements dans le web de données”. In : *21es Journées Francophones d’Ingénierie des Connaissances (IC 2010)*, <http://www.ic2010.mines-ales.fr/>. Ecole des Mines d’Alès. 2010, p. 69–80.
- [177] Subodh VAID, Christopher B. JONES, Hideo JOHO et Mark SANDERSON. “Spatio-textual Indexing for Geographical Search on the Web”. In : *Proceedings of the 9th International Conference on Advances in Spatial and Temporal Databases*. SSTD’05. Angra dos Reis, Brazil : Springer-Verlag, 2005, p. 218–235. DOI : 10.1007/11535331_13. URL : http://dx.doi.org/10.1007/11535331_13.
- [178] Andreas VLACHOS. “Semi-supervised learning for biomedical information extraction”. Thèse de doct. Computer Laboratory, University of Cambridge, 2010.
- [179] Ellen M. VOORHEES et Donna K. HARMAN. *TREC : Experiment and Evaluation in Information Retrieval (Digital Libraries and Electronic Publishing)*. The MIT Press, 2005. ISBN : 0262220733.
- [180] David RF WALKER, Ian A NEWMAN, David J MEDYCKYJ-SCOTT et Clive LN RUGGLES. “A system for identifying datasets for GIS users”. In : *International Journal of Geographical Information Science* 6.6. Taylor & Francis, 1992, p. 511–527.

BIBLIOGRAPHIE

- [181] J. WANG, K.Y. CHEN, E. KAYIS, G. GALLEGRO, J.L.B. GUERRERO, R. WANG et S.K. JAIN. *Tree-based regression*. US Patent App. 13/528,972. 2013. URL : <https://www.google.com/patents/US20130346033>.
- [182] Sujing WANG et Christoph F. EICK. “A Polygon-based Clustering and Analysis Framework for Mining Spatial Datasets”. In : *Geoinformatica* 18.3. Hingham, MA, USA : Kluwer Academic Publishers, juil. 2014, p. 569–594. ISSN : 1384-6175. DOI : 10.1007/s10707-013-0190-2. URL : <http://dx.doi.org/10.1007/s10707-013-0190-2>.
- [183] Wei WANG et Kathleen STEWART. “Spatiotemporal and semantic information extraction from Web news reports about natural hazards”. In : *Computers, environment and urban systems* 50. Elsevier, 2015, p. 30–40.
- [184] William E WINKLER. “The state of record linkage and current research problems”. In : *Statistical Research Division, US Census Bureau*. Citeseer. 1999.
- [185] Krist WONGSUPHASAWAT, Catherine PLAISANT, Meirav TAIEB-MAIMON et Ben SHNEIDERMAN. “Querying event sequences by exact match or similarity search : Design and empirical evaluation”. In : *Interacting with computers* 24.2. Elsevier, 2012, p. 55–68.
- [186] Zhibiao WU et Martha PALMER. “Verbs Semantics and Lexical Selection”. In : *Proceedings of the 32Nd Annual Meeting on Association for Computational Linguistics*. ACL '94. Las Cruces, New Mexico : Association for Computational Linguistics, 1994, p. 133–138. DOI : 10.3115/981732.981751. URL : <http://dx.doi.org/10.3115/981732.981751>.
- [187] Ugan YASAVUR, Reza AMINI, Christine L LISETTI et Naphtali RISHE. “Ontology-Based Named Entity Recognizer for Behavioral Health.” In : *FLAIRS Conference*. 2013.
- [188] Yu Z. “High accuracy postal address extraction from web pages”. In : 2007.
- [189] Guoqiang ZHANG, B Eddy PATUWO et Michael Y HU. “Forecasting with artificial neural networks : The state of the art”. In : *International journal of forecasting* 14.1. Elsevier, 1998, p. 35–62.
- [190] GuoDong ZHOU et Jian SU. “Named Entity Recognition Using an HMM-based Chunk Tagger”. In : *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. ACL '02. Philadelphia, Pennsylvania : Association for Computational Linguistics, 2002, p. 473–480. DOI : 10.3115/1073083.1073163. URL : <http://dx.doi.org/10.3115/1073083.1073163>.