Content and data linking leveraging ontological knowledge in data journalism

Cheikh Brahim El Vaigh

UNIVERSITE BRETAGNE LOIRE / MATHSTIC

UNIVERSITÉ DE RENNES 1

THÈSE DE DOCTORAT DE

L'UNIVERSITÉ DE RENNES 1
COMUE UNIVERSITÉ BRETAGNE LOIRE

ÉCOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Informatique*

Par

« Cheikh Brahim EL VAIGH »

« Content and data linking leveraging ontological knowledge in data journalism »

Thèse présentée et soutenue à Rennes, le 7 janvier 2021
Unité de recherche : Inria Rennes

Rapporteurs avant soutenance :

Nathalie PERNELLE	Professeur à Sorbonne Université Paris Nord
Xavier TANNIER	Professeur à Sorbonne Université

Composition du Jury :

Examinateurs :	Peggy CELLIER	Maître de conférences à INSA Rennes
	Amedeo NAPOLI	Directeur de recherche CNRS
	Nathalie PERNELLE	Professeur à Sorbonne Université Paris Nord
	Sophie ROSSET	Directeur de recherche CNRS
	Xavier TANNIER	Professeur à Sorbonne Université
Dir. de thèse :	Guillaume GRAVIER	Directeur de recherche CNRS
Co-dir. de thèse :	François GOASDOUÉ	Professeur à Université de Rennes 1
	Pascale SÉBILLOT	Professeur à INSA Rennes

Acknowledgement

I would like to give my sincere thanks to my supervisors, François Gouasdoué, Guillaume Gravier and Pascale Sébillot for their feedback and outstanding advices not only during the three years of my thesis, but also after I finished it. Thank you for everything!

Moreover, I would like to thank my reviewers and jury members, Peggy Cellier, Amedeo Napoli, Nathalie Pernelle, Sophie Rosset and Xavier Tannier, for the evaluation of my work, the valuable discussions and their kind words during and before the defense. In addition, I would like to thank Peggy Cellier and Emmanuel Morin for being in my CSID and following my thesis all over the three years.

I would also like to thank Inria for funding my thesis within the IPL iCooda, as well as IRISA for welcoming me in the Lab. Furthermore, I would like to thank iCoda members for the exceptional moments we had together during the different meetings.

A special thanks to Linkmedia members for their kindness, the fantastic moments we spent together and all the interesting discussions we used to have. Many thanks to my new team members (Shaman) for their support.

Last but not least, I would like to thank my family and friends for their unconditional support throughout this particular period of my life and especially during the first year, when my mom passed away.

Table of Contents

Résumé en français **9**

1 Introduction **15**

2 Background **21**
2.1 Knowledge Representation 22
2.2 Word Embeddings 25
2.3 Knowledge Bases Embeddings 26
2.4 Jointly Embedding Words and KBs 27
2.5 Conclusion 28

3 Collective Linking With WSRM **29**
3.1 Related Work 30
3.1.1 Entity-By-Entity Linking 31
3.1.2 Collective Entity Linking 32
3.1.3 Limits of Entity Linking Techniques 34
3.2 Collective Linking with KB Semantics 35
3.2.1 Entities Relatedness 36
3.2.2 Candidate Entities Generation 37
3.2.3 Local Mention-Entity Score 38
3.2.4 Supervised Collective Entity Linking 38
3.3 Experiments 39
3.3.1 Experimental Setup 39
3.3.2 Ablation Study 41
3.3.3 Comparison to State-of-the-Art Approaches 42
3.4 Conclusion 45

4 Collective Linking With ASRMPm **47**
4.1 Requirements for a Well-founded Entity Relatedness Measure 48
4.2 The Path-based Weighted Semantic Relatedness Measure 51

4.3 Linking with entity relatedness measure . . . 56
4.3.1 Knowledge Base . . . 57
4.3.2 Supervised Entity Selection . . . 57
4.4 Experiments . . . 59
4.4.1 Implementation Details . . . 59
4.4.2 Comparing Fuzzy Logic Aggregators . . . 60
4.4.3 Comparing Classifiers . . . 60
4.4.4 Comparing Aggregation Strategies . . . 61
4.4.5 Entity Relatedness Results . . . 62
4.4.6 Comparison of entity linking systems . . . 65
4.4.7 Beyond Figures... . . . 65
4.5 Conclusion . . . 66

5 Knowledge Graphs Alignment 68
5.1 Related Work and Notations . . . 69
5.1.1 Overview of existing work . . . 69
5.1.2 Background notations and technical details . . . 71
5.2 AlignD . . . 73
5.2.1 Measuring Embedding Alignment . . . 74
5.2.2 AlignD Objective Function and Algorithm . . . 76
5.2.3 Automatic Seed Replacement . . . 78
5.3 Experiments . . . 78
5.3.1 Experimental Setup . . . 79
5.3.2 Comparison to State-of-the-Art Approaches using a Ground-Truth Seed . . . 81
5.3.3 Impact of the seed size on performance . . . 82
5.3.4 Corrupted Seed Modeling . . . 83
5.3.5 Automatically Extended Seed . . . 84
5.3.6 Automatically Generated Seed . . . 84
5.4 Conclusion . . . 86

6 Conclusion 89

Conclusion 89
6.1 Summary of the Contributions . . . 89

6.2 Perspectives 90
6.2.1 General Entity Relatedness Measures 91
6.2.2 Word and Entity Joint Embedding for CEL 91
6.2.3 Indirect Criterion for Word and Entity Joint Embedding 92
6.2.4 Impact of Entity Alignment on CEL 92
6.2.5 Local Global Entity Linking 92

Bibliography **95**

Résumé en français

Les ordinateurs jouent un rôle important dans notre vie quotidienne. Ils sont en particulier utilisés pour produire et traiter du contenu numérique, tel que des journaux en ligne. Il est donc important que les ordinateurs soient (en quelque sorte) capables de comprendre le langage naturel utilisé dans les documents textuels afin de pouvoir les traiter avec précision.

Les techniques de traitement automatique des langues (TAL) visent à programmer des ordinateurs afin qu'ils puissent traiter et analyser une grande quantité de données textuelles. Plus précisément, le TAL permet à l'ordinateur de comprendre dans une certaine mesure la signification des mots présents dans une phrase. Par ailleurs, le Web sémantique aide à représenter des données du monde réel (des entités et leurs relations) dans une base de connaissances (BC), accessible par des machines, et qui encode la sémantique des relations entre les entités de cette BC.

Dans cette thèse, nous combinons les techniques du Web sémantique et du TAL pour permettre une meilleure compréhension du texte. Par exemple, étant donné la phrase " Steve Jobs est un co-fondateur d'Apple.", la mention " Apple" – mot faisant référence à une entité (instance d'un concept d'une BC) – peut être utilisée pour le fruit " Apple", ou la société " Apple Inc.", ambiguïté qui peut être résolue en trouvant le sens exact du mot " Apple" en fonction de son contexte. Utiliser les relations sémantiques entre les entités d'une BC aide à trouver l'entité qui contient " Apple" dans son nom et qui est la plus susceptible d'être liée à l'entité nommée " Steve Jobs". Trouver la signification exacte des mentions, comme dans la phrase précédente, est appelé liaison d'entités ou *entity linking*. C'est une étape cruciale vers la compréhension automatique du contenu des documents textuels.

La tâche d'*entity linking* vise donc à trouver les mentions d'entités dans un document textuel et à relier chaque mention à une entité unique dans une BC. Les techniques du TAL et du Web sémantique sont essentielles pour résoudre une tâche d'*entity linking*. Les premières sont utilisées pour repérer les mentions d'entités dans un texte, tandis que les secondes servent à trouver la bonne entité pour chaque mention détectée dans le texte. Extraire automatiquement des mentions d'entités à partir d'un texte et les relier à leurs

entités correspondantes dans une *BC* est utile pour divers objectifs, tels que la recherche sémantique, c'est-à-dire la recherche d'entités fondée sur leur signification plutôt que sur leurs noms, ou encore l'extraction de connaissances qui vise à récupérer des faits présents dans un texte non structuré ou au peuplement d'une *BC* grâce à des faits trouvés dans un texte .

La littérature du domaine de l'*entity linking* distingue deux catégories principales, selon que les mentions dans un document sont liées indépendamment les unes des autres ou collectivement. La première catégorie lie une mention (référence à une entité) dans un texte à l'entité la plus similaire dans une *BC* moyennant une mesure de similarité, telle que la similarité cosinus, entre leurs représentations dans un espace multidimensionnel, tandis que le liage d'entités collectif (*collective entity linking*) utilise en outre les propriétés de la *BC* pour sélectionner le meilleur appariement global d'entités pour toutes les mentions dans le texte, en se fondant sur les relations entre les entités au sein de la *BC*. Les liens entre les entités sont estimées à l'aide d'une *mesure de similarité sémantique*, qui reflète l'affinité des entités de la *BC*. Dans cette thèse, nous nous concentrons sur l'*entity linking* collectif car il intègre dans le processus de liage les relations entre entités au sein d'une *BC*, ce qui permet, dans une certaine mesure, de bénéficier de la sémantique des *BC*. Nous soutenons la thèse qu'utiliser des *BC* RDF, qui ont une sémantique claire et précise, car elles représentent des concepts et des relations avec une signification claire pour un humain, permet d'améliorer l'état de l'art du liage d'entités collectif.

Les techniques d'*entity linking* utilisent généralement les informations contenues dans des *BC*, telles que les noms et les descriptions d'entités, ainsi que la structure de la *BC* utilisée pour concevoir des mesures sémantiques entre entités. Un grand nombre de travaux utilisent Wikipedia [1] [1, 9, 16, 34, 28, 59, 43], exploitant les interconnexions entre ses pages Web. Ces interconnexions sont implicitement capturées à l'aide du graphe d'hyperliens de Wikipedia. Il n'y a cependant qu'une sémantique plutôt faible derrière ces interconnexions, qui informent simplement de l'existence d'une relation non spécifiée entre deux entités *via* le graphe d'hyperliens. Pour contourner ce problème, quelques approches utilisent les *BC* fondées sur le Resource Description Framework (RDF), telles que BaseKB, DBpedia ou YAGO, en lieu et place de Wikipedia [75, 35, 53]. Elles bénéficient ainsi théoriquement de l'utilisation du modèle de données RDF pour la représentation des connaissances, notamment pour exploiter la sémantique précise des entités de la base de connaissance (par exemple les types) et de leurs relations (par exemple les noms des propriétés et

1. https://www.wikipedia.org/

leurs cardinalités). Les approches utilisant des BC fondées sur RDF dans le contexte du collectif *entity linking* n'exploitent cependant pas réellement la sémantique des relations au sein d'une BC, se limitant dans le meilleur des cas à un indicateur de l'existence d'une relation entre deux entités [46].

Pour exploiter pleinement la sémantique des BC RDF, nous définissons une *mesure sémantique* (WSRM) qui permet de mieux prendre en compte les relations entre les entités au sein de la BC. WSRM va au-delà de l'existence de relations entre deux entités et propose de pondérer le lien entre deux entités en tenant compte du nombre de relations qu'elles ont en commun dans la BC. Pour améliorer l'état de l'art, nous proposons également une technique d'*entity linking* collectif qui se fonde sur WSRM. Les principaux ingrédients de notre technique sont, d'une part, la popularité Wikipedia (voir Eq. 3.2) et la similarité cosinus entre mentions et entités candidates – à l'aide du modèle de plongement de mots skip-gram – et, d'autre part, un score global calculé sur toutes les entités candidates choisies pour les différentes mentions à l'aide de notre mesure WSRM. Nous montrons à travers une validation expérimentale qu'il est faisable et bénéfique de prendre en compte la sémantique des relations entre les entités d'une BC RDF.

Ces résultats, en plus de montrer l'intérêt d'utiliser des BC RDF, ouvrent de nouvelles perspectives pour prendre en compte la richesse et l'expressivité des BC structurées pour le liage collectif d'entités. En particulier, utiliser des BC RDF ouvre la porte pour exploiter pleinement leur sémantique à l'aide de mécanismes de raisonnement. Alors que dans notre première contribution, nous nous sommes limité aux relations directes entre entités dans une BC RDF, une extension possible est de considérer également les relations indirectes entre entités, par exemple les chemins de longueur supérieure à un entre deux entités. Néanmoins, cette extension pose plusieurs défis tels que le contrôle de la sémantique des chemins de longueur supérieure à un, ou encore la réduction de la complexité induite par cette extension, car le nombre de chemins indirects explose rapidement.

Pour soulever les défis posés par l'extension citée précédemment, nous étudions les propriétés des *mesures sémantiques bien fondées* qui utilisent la sémantique des BC RDF, dans le cadre de l' *entity linking* collectif. En particulier, nous supposons qu'en plus de permettre une amélioration significative de l'état de l'art, une mesure *bien fondée* devrait, dans la mesure du possible, répondre aux trois critères suivants : **(R1)** elle doit avoir une *sémantique claire* pour que les décisions de l'*entity linking* puissent être facilement comprises ou expliquées, s'appuyant sur une base de connaissances sémantique (RDF ou OWL, par opposition à Wikipedia) et évitant tout réglage de paramètres qui est difficile à

définir par les utilisateurs finaux ; **(R2)** elle doit avoir un *coût de calcul raisonnable* pour être d'un intérêt pratique, et **(R3)** elle doit être *transitive* en encodant le mécanisme de composition de relations, pour capturer le fait que les entités peuvent être liées directement ou indirectement dans une BC, par exemple *via* des *chemins* dont la taille est supérieure à un. Le dernier critère **(R3)** est crucial car il permet d'encoder des liens implicites entre les entités. Par exemple, si X *travaillePour* Y et Y *EstDans* Z alors, le chemin de X à Z encode implicitement X *travailleDans* Z, ce qui correspond à une information non stockée dans la BC pouvant être capturée par les mesures qui respectent **(R3)**. Nous définissons ainsi une famille de mesures sémantiques qui bénéficient de ces propriétés, dans le but d'améliorer l'état de l'art du liage d'entités collectif.

Nous proposons donc de définir une nouvelle *mesure sémantique bien fondée* (ASRMP_m), qui étend les mesures précédentes – dont WSRM – en tenant compte des relations indirectes entre les entités d'une BC, à savoir les chemins de taille m, $m > 1$. Nous montrons la faisabilité de l'incorporation des chemins indirects tout en conservant la même complexité et une sémantique claire, en exploitant pour ce faire des agrégateurs flous. Nous validons aussi expérimentalement l'intérêt de ASRMP_m pour le *collective entity linking*, où les chemins de longueur $m = 2$ et $m = 3$ apportent une amélioration par rapport aux mesures de l'état de l'art qui utilisent soit uniquement des connexions directes entre entités [24], soit des chemins directs et indirects [36]. En théorie, la complexité de ASRMP_m est inversement proportionnelle à la longueur des chemins. Nous montrons toutefois que l'intérêt de cette mesure est limité aux chemins utiles, c'est-à-dire ceux d'une longueur allant jusqu'à $m = 3$; en pratique en effet les chemins plus longs ($m > 3$) n'ajoutent que du bruit. $ASRMP_m$ permet de bénéficier de la sémantique des BC RDF, et introduit du raisonnement basique en exploitant les chemins indirects. En conclusion, nous pouvons dire que des mesures sémantiques bien fondées permettent d'améliorer la précision de l'*entity linking* collectif, ce dernier facilitant à son tour l'exploration et l'analyse de documents textuels.

Traditionnellement, l'*entity linking* est effectué en s'appuyant sur une BC statique, ayant des entités et des relations fixes. Une telle BC restreint l'*entity linking* car elle peut ne pas contenir l'entité correcte correspondant à une mention donnée présente dans un texte. Par ailleurs, les *mesures sémantiques* utilisées pour le liage d'entités collectif étant définies à partir des relations existant entre les entités de la BC utilisée, la qualité de ces mesures est donc directement liée au contenu de la base. Il est donc crucial de disposer d'une base de connaissance suffisamment riche et à jour, ce qui peut se faire en

alignant différentes BC et en les mettant continuellement à jour. L'*alignement de BC* ou *alignement d'entités* permet de réaliser cet objectif.

Les méthodes conventionnelles d'alignement d'entités lient les entités en fonction de leurs caractéristiques symboliques, telles que leurs noms, leurs types [64]. Néanmoins, ces approches souffrent d'un problème d'hétérogénéité sémantique entre les diverses BC, en particulier des schémas et conventions de nommage différents. Plus récemment, des techniques de plongement de BC ont été utilisées pour contourner ce problème. L'idée est d'apprendre les représentations vectorielles d'entités et de relations d'une BC, qui reflètent la sémantique de la base. L'alignement d'entités est résolu en apprenant conjointement (en même temps) le plongement des deux BC à aligner dans le même espace sémantique, en réduisant la distance entre les entités qui sont les mêmes dans les deux BC. L'alignement d'entités fondé sur les techniques de plongement s'appuye sur un ensemble d'entités alignées fourni comme *graine initiale* (*seed*) pour guider le plongement dans l'espace conjoint et garantir que les entités devant être alignées se trouvent à proximité. Cette graine, qui assure donc la correspondance entre les entités des deux BC pour quelques entités, est aujourd'hui le goulot d'étranglement majeur des techniques de l'état de l'art : sa création, requérant des experts, est coûteuse, et sa taille et sa qualité ont un impact substantiel sur la précision des techniques de l'alignement d'entités.

Pour contourner les limites des techniques d'alignement d'entités, nous proposons un nouveau critère, nommé *AlignD*, pour apprendre les plongements conjoints de BC et résoudre la tâche de l'alignement d'entités. Les approches traditionnelles cherchent à minimiser explicitement la distance entre les entités alignées de la graine, généralement en se fondant sur la norme L_2. Nous définissons pour notre part plutôt un critère qui cherche à maximiser globalement la corrélation entre les dimensions de l'espace de plongement pour les entités alignées dans la graine, réduisant ainsi indirectement la distance entre ces entités. Ce critère permet d'ignorer les erreurs potentiellement présentes dans la graine, puisque la graine est ici considérée globalement, ce qui conduit à une technique d'alignement d'entités robuste. Nous montrons expérimentalement que la robustesse de notre critère global permet de remplacer la graine vérité-terrain par une graine générée automatiquement sans réduire la qualité finale de l'alignement, excluant ainsi le besoin d'experts humains pour le processus d'alignement d'entités.

Plusieurs perspectives s'offrent à l'issue des travaux discutés au sein de ce mémoire. La mesure $ASRMP_m$ que nous avons proposée a été conçue pour la tâche du liage d'entités collectif. Néanmoins, les mesures sémantiques entre entités peuvent être utilisées pour

d'autres tâches telles que la visualisation de *BC* RDF, l'apprentissage de métrique, ou encore la recherche d'entités, qui peut être utilisée dans un moteur de recherche sémantique, et où l'idée est de trouver, pour une entité donnée, les entités les plus similaires présentes dans une *BC*. Nous estimons donc qu'une analyse approfondie des mesures sémantiques à la lumière des critères que nous avons définis dans ce mémoire peut profiter aux tâches susmentionnées. Par ailleurs, dans notre solution pour l'*entity linking* collectif, nous nous sommes principalement limités aux mesures sémantiques. Cependant, les techniques de plongement peuvent être utilisées pour résoudre tous les problèmes de liaison d'entités et de mentions, en se fondant sur un plongement conjoint de mots et d'entités. Nous avons utilisé le modèle skip-gram pour apprendre le plongement de mots et TransE pour le plongement de *BC*, mais un modèle d'*embedding* conjoint mot-entité pourrait être plus avantageux pour l'*entity linking*. Enfin, l'alignement d'entités fournit une solution simple pour lier des entités de plusieurs *BC* et peut être exploité pour effectuer de l'*entity linking*. Imaginons que nous ayons lié un corpus de texte à une *BC* BC_1 ; nous pouvons aligner cette *BC* avec une nouvelle *BC* BC_2, ayant ainsi indirectement lié le corpus de texte à BC_2 en utilisant le processus d'alignement. Cette solution permet d'être plus flexible dans le choix des *BC* utilisées pour l'*entity linking*. Une telle solution n'est pas exempte d'erreurs, et se pose donc la question de son efficacité. Une analyse approfondie des techniques d'alignement d'entités permettra de montrer dans quelle mesure l'alignement d'entités est bénéfique pour l'*entity linking* collectif indirect.

Chapter 1

Introduction

Computers play an important role in our everyday life. In particular, they are used to process and construct digital textual contents, such as the content of online newspapers. Hence, it is mandatory for computers to read and (kind of) understand the natural language used in textual documents in order to accurately process this digital content.

Natural language processing (NLP) techniques aim at programming computers so they can process and analyze large amounts of natural language data. More specifically, NLP allows computer to understand (to a certain extent) word meaning in a sentence.

On the other hand, Semantic Web helps representing data from real world (entities and their relations) in a knowledge base (KB), which is manageable by machines, and encodes the semantics of entities interrelationships.

In this thesis, we combine Semantic Web and NLP techniques to improve text understanding with comparison to using only NLP techniques, which may not efficiently use the relations between words in a sentence. For example, given the sentence “Steve Jobs is a co-founder of Apple.”, the mention "Apple"—word referring to an entity—can be used for the fruit "Apple", or the company "Apple, Inc", this ambiguity can be solved by finding the exact meaning of the word "Apple" based on its context. Making use of semantic relations between entities in a KB will help to find the entity that contains "Apple" in its name and that is more likely to be linked to the entity named "Steve Jobs". Finding the exact meaning of the mentions, as in the previous sentence is known as *entity linking*. It is a crucial step toward automatically understanding the meaning of textual documents.

The task of entity linking therefore aims at finding the mentions of entities in a textual document and linking each mention to a unique referent in a KB. Both natural language processing and semantic web techniques are instruments for entity linking. The former are used to retrieve mentions from text, while the later are used to compute a unique referent for each mention previously found in the text. Automatically extracting mentions of entities from unstructured text and linking them to their corresponding entities in a structured KB serves many purposes such as: semantic search that is, searching entities

based on their meaning rather than their names; knowledge extraction which aims at retrieving facts from unstructured texts or KB population meaning, injecting facts found in a text into a KB.

Entity linking is however a non-trivial task because entity mentions are usually ambiguous. For example, given the previous sentence “Steve Jobs is a co-founder of Apple.”, the mention “Steve Jobs” refers to distinct entities within the Freebase KB [6]: a person, a book, a film, etc. Conversely, an entity may be mentioned in various forms where, for instance, the Freebase entity “Steve Jobs” may appear in a document as “Steve Jobs”, “Steve Paul Jobs”, “Steve P. Jobs”, “Steve” and “Jobs”, etc. Thus entity linking raises several challenges such as mentions ambiguity which means that the same mention can refer to different entities; entity name variants, since the same entity can be mentioned using different string surface forms; indirect mentions, as an entity can be mentioned with a string surface form that is very different from the entity name e.g., "Steve jobs" and "the CEO of Apple".

The literature for entity linking distinguishes two main approaches, depending on whether mentions within a single document are linked to entities independently one from another or collectively at once. The former links every mention in text to the most similar entity in a KB based on their similarity e.g., cosine similarity between their representations in a multidimensional space, while collective linking further uses the KB to select the best global set of entities for all the mentions in text, based on entity interrelationships as embedded in the KB. Entity interrelationships are estimated by the mean of a so-called *entity relatedness measure*, i.e., affinity between entities in the KB. In this thesis, we focus on the collective entity linking as it incorporates entity interrelationships in the linking process and allows benefitting from KBs semantic to some extent.

Entity linking techniques usually make use of the information within the KB—e.g., entity names and descriptions whenever they are available—and of its structure to devise an entity relatedness measure. A large body of work use Wikipedia [1], e.g., [1, 9, 16, 34, 28, 59, 43], exploiting its entity interconnections. The latter are implicitly captured by means of the hyperlink graph of Wikipedia web pages. There is however only loose semantics behind those interconnections, which prevents from fully taking advantage of KB semantics to better exploit the relationship between entities. To skirt this issue, a few approaches use Resource Description Framework (RDF) structured KBs, such as BaseKB, DBpedia or YAGO, instead of Wikipedia [75, 35, 53]. They thus theoretically benefit from the

1. https://www.wikipedia.org/

use of the formal RDF data model for knowledge representation, in particular to exploit the precise semantics of the KB entities (e.g., types) and of their interrelationships (e.g., names and cardinalities), while Wikipedia can just inform about the existence of some unspecified relation between two entities through its hyperlink graph. Approaches that use the RDF KB structure in the context of collective entity linking however do not exploit the semantics —clear meaning— of the relations within the KB, limiting themselves in the best case to a binary indicator of whether a relation exists or not between two entities [46]. They moreover use costly algorithms for collective entity linking, which makes them poorly suitable for large scale document engineering. In this thesis, we show the benefit of using to the extent possible —direct and indirect— entity interrelationships in RDF KBs for the collective linking.

To capitalize on the precise semantics of RDF KBs, we first contribute by the definition of a fine-grained *weighted semantic relatedness measure* (WSRM), which we design to better capture the description of the semantic interrelations between entities that are available in structured RDF KBs. WSRM goes beyond the existence of some relations between two entities and propose to weight them accounting for the number of relations they have in common in the reference KB. Furthermore, we provide a *lightweight* collective entity linking technique on top of the proposed entity relatedness measure which is more suitable for linking large scale documents with comparison to the standard collective entity linking techniques. Finally, we study the properties of entity relatedness measures and provide the requirements for a well-founded entity relatedness measure that capitalizes the most on the precious semantics of RDF KBs for the collective linking. We thus define a family of entity relatedness measures that meet these requirements, leading to a new state-of-the-art entity relatedness measures for the collective linking.

Traditionally, entity linking is performed on a static KB with a fixed number of entities and relations. Such a KB restricts the linking as it may not contain the correct entity for a given mention. Moreover, the entity relatedness measures used for the collective linking are defined based on the relations between the entities of a KB, the quality of these measures being thus tied to the content of the KB used. Thus, it is crucial to connect—align— different KBs and continually update them by exchanging their knowledge. This process of connecting KBs allows to build a more rich and general one, which will allow to build high quality entity linking techniques. Thus KBs *alignment*, a.k.a. *entity alignment*, allows to benefit from any existing KBs in order to build a richer KB which can be used for entity linking.

Entity alignment consists in finding entities from different KBs that are the same in order to link them. Entity alignment is challenging, since entities, while being the same, may have different names, attributes in different KBs which raises semantic ambiguity. On the other hand, different entities may have the same names, which makes the alignment hard based only on entities attributes.

Conventional methods for entity alignment match entities based on their symbolic features, such as their name, type or attributes in general [64]. Nevertheless, symbolic features suffer from the semantic heterogeneity issue between different KBs, in particular different schemas and naming conventions. For example in BaseKB, there are several entities with the same name "Steve Jobs"; matching them based on their names only, will probably fail. More recently, KBs embedding techniques have been used to bypass the semantic heterogeneity. The idea is to learn vector representations of entities and relations in KBs, which reflect entity semantics. Entity alignment is therefore solved by jointly learning the embedding of two KBs in the same embedding space, reducing the distance between similar entities. Entity alignment based on KG embedding techniques additionally makes use of a prior set of aligned entities as a *seed* to guide the embedding and ensure that aligned entities lie close in the embedded space. This seed, which provides the correspondence between entities of the two KBs for a few entities, is today the major bottleneck of state-of-the-art techniques. Creating the seed is costly, as it is provided by experts. Furthermore, its size and quality has a substantial impact on the alignment accuracy.

To address the limits of entity alignment techniques, we propose a novel criterion to learn joint embeddings of KBs for entity alignment. Traditional approaches seek to explicitly minimize the distance between aligned entities from the seed, typically based on the L_2-norm in the embedded space. We rather define a criterion that globally seeks to maximize the correlation across dimensions of the embedding space for aligned entities in the seed, thus indirectly lowering the distance between these entities. This criterion allows to ignore small errors in the seed, as we consider the seed globally, which leads to a robust entity alignment technique. We experimentally show that the robustness of our global criterion, enables replacing ground-truth seed with automatic seed generation, entirely excluding human input from the alignment process.

The current thesis is part of the iCODA project [2] that aims at helping journalists exploring efficiently large archives of newspapers. iCODA seeks to design tools that will help

2. http://project.inria.fr/icoda

journalists querying data from different sources e.g., different RDF KBs; visualizing data with human friendly interfaces that will facilitate the collaboration of different journalists; and finally linking archives content to a reference KB which will allow to enrich the KB at hand with the content of the archives. This project brings together four Inria teams with complementary expertise from several areas of research namely CEDAR, GraphIK, ILDA and Linkmedia. This thesis, is mainly concerned by linking the content of an archive to a reference KBs. We Hence, seek to apply entity linking to a large archive of a regional newspaper using a KB of specific places, companies, public bodies and people.

This thesis is concerned with the problem of building links between content and KBs. We advocate that capitalizing on the precious semantics of RDF KBs leads to better performance for the tasks of entity linking and entity alignment.

This document is composed of three parts organized as follows:

— First, Chapter 2 gives a brief background about knowledge representation and embedding techniques for both words and KBs. More specifically, we further explain the RDF data model and the representations we are using for entity linking and entity alignment.

— Then, we introduce the task of entity linking in Chapter 3. We describe our lightweight collective entity linking where our measure WSRM is used to compute distance between entities. This measure which is our first contribution, goes beyond the existence of a relation between two entities, and proposes to weight entities in a KB accounting for their interrelations. Moreover, we propose in Chapter 4 to define the requirements for a well-founded entity relatedness measure by identifying the quality of entity relatedness measures from the literature for collective entity linking. Furthermore, we contribute by the definition of a family of entity relatedness measures that we show to be more suitable for collective entity linking with comparison to entity relatedness measure in the literature.

— Finally, Chapter 5 presents our contribution for the task of entity alignment which consists in a global criterion for entity alignment, designed to perform entity matching without any prior knowledge between the KBs under consideration. We define a robust entity alignment technique that globally maximize the correlation across dimensions of the embedding space for aligned entities rather than directly minimizing the cosine similarity between these entities. Our solution ultimately enable to exclude expert annotations from the process of entity alignment.

Chapter 2

Background

We deal with two aspects of linking in this thesis : entity linking, performed between a document and a reference KB; and entity alignment which links the entities of two different KBs. In both cases, the representation of facts within a KB is crucial as it allows defining the semantics of the KB, in particular relationships between entities.

Two types of KBs are used so far in the context of linking, namely Wikipedia-like KBs and ontological KBs. Relations between entities within a Wikipedia-like KB —that use hyperlinks to define relations between two entities— are implicitly captured by the hyperlink graph of Wikipedia web pages. There is however only loose semantics behind these interconnections as they just inform about textual hyperlinks between two web pages, regardless to their importance to these web pages. By contrast to Wikipedia-like KBs, in an ontological KBs, data representation is based on a formal data model such as the popular graph data model RDF [72] and allow defining a rich and clear semantics for the relations between entities. For instance, the semantics of the relations between entities is important as it allows defining a semantic distance between entities e.g., entity relatedness measure, which is an ingredient of the collective entity linking. However, the semantics of entity relatedness measures is based on the facts representation in the KB at hand. We advocate for RDF KBs to solve the task of collective linking. RDF data model which is the cornerstone of RDF KBs is described in Sec. 2.1.

The definition of a reference KB is not sufficient to perform linking since mentions in a text are linked to entities in a KB based on their distance. Therefore, it is mandatory to define this distance, and one straightforward solution is the cosine similarity. The idea is to encode the names of entities in a KB and the mentions in a text as points in a multidimensional space —word embedding— and use their similarity as an ingredient for the linking. Following many pieces of work in the literature of entity linking, we account for the skip-gram embedding model [50] to compute the similarity a.k.a. *local score* between mentions in text, and their associated entities in a KB. However, we are not limited to skip-gram, and any recent embedding model [57, 5, 58, 18] can be used to devise the local

score. We further detail these word embedding models in Sec. 2.2.

Entity linking techniques perform the linking to a static KB which may prevent from correctly linking mentions in text. To skirt this issue, entities of different KBs are matched based on their distance in a multidimensional space, allowing to gather entities from different KBs in one KB. This time, the points in the multidimensional space are entities and such a process is called KB embedding. We build our entity alignment system on the TransE [7] KB embedding technique which we develop in Sec 2.3.

In the context of entity linking, words and KBs embedding are very often learned independently regardless of the relation words and entities may have. Since, mentions in a text are necessarily referring to entities in a KB, learning a joint representation of words and entities allow to incorporate the links between mentions and entities in the joint embedding space. Hence, we present in Sec 2.4 the prominent joint embedding models used for the task of entity linking.

2.1 Knowledge Representation

Knowledge representation is the artificial intelligence (AI) field dedicated to formally representing information about the real world. It allows to rethink data modeling, making the information more understandable by a machine. In particular, good graph data model facilitate solving complex AI tasks such as semantic search [4] and information extraction [37, 38, 39]. In our case, knowledge representation is used to improve the way we solve the problems of entity linking.

The Resource Description Framework (RDF) by the W3C [72] is used to express both data and domain knowledge in a structured KB. RDF is the cornerstone of semantic web applications, whose emblematic incarnation is the linked open data cloud [1] and several KBs are based on RDF such as Yago [2], DBpedia [3] or Wikidata [4].

Formally, an RDF KB is a set of (s, p, o) *triples* [71], each describing its *subject* s with the *property* p whose value —called *object*— is o. An RDF KB can be seen as a graph made of $s \xrightarrow{p} o$ edges. Triples are used to state facts and domain knowledge using *special properties* from the RDF standard, as shown in Table 2.1 and exemplified next.

1. `https://lod-cloud.net`
2. `https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago`
3. `https://wiki.dbpedia.org/`
4. `https://www.wikidata.org/wiki/Wikidata:Main_Page`

RDF fact	Triple notation
Class assertion	$(s, \texttt{type}, o)$
Property assertion	(s, p, o) with $p \notin \{\texttt{type}, \texttt{subClassOf}, \texttt{subPropertyOf}, \texttt{domain}, \texttt{range}\}$
RDF knowledge	Triple notation
Subclass	$(s, \texttt{subClassOf}, o)$
Subproperty	$(s, \texttt{subPropertyOf}, o)$
Domain typing	$(s, \texttt{domain}, o)$
Range typing	$(s, \texttt{range}, o)$

Table 2.1 – RDF triples for facts and knowledge.

Rule [74]	Entailment rule
rdfs2	$(p, \texttt{domain}, c)$, $(s, p, o) \rightarrow (s, \texttt{type}, c)$
rdfs3	$(p, \texttt{range}, c)$, $(s, p, o) \rightarrow (o, \texttt{type}, c)$
rdfs5	$(p_1, \texttt{subPropertyOf}, p_2)$, $(p_2, \texttt{subPropertyOf}, p_3) \rightarrow (p_1, \texttt{subPropertyOf}, p_3)$
rdfs7	$(p_1, \texttt{subPropertyOf}, p_2)$, $(s, p_1, o) \rightarrow (s, p_2, o)$
rdfs9	$(c_1, \texttt{subClassOf}, c_2)$, $(s, \texttt{type}, c_1) \rightarrow (s, \texttt{type}, c_2)$
rdfs11	$(c_1, \texttt{subClassOf}, c_2)$, $(c_2, \texttt{subClassOf}, c_3) \rightarrow (c_1, \texttt{subClassOf}, c_3)$
ext1	$(p, \texttt{domain}, c_1)$, $(c_1, \texttt{subClassOf}, c_2) \rightarrow (p, \texttt{domain}, c_2)$
ext2	$(p, \texttt{range}, c_1)$, $(c_1, \texttt{subClassOf}, c_2) \rightarrow (p, \texttt{range}, c_2)$
ext3	$(p, \texttt{subPropertyOf}, p_1)$, $(p_1, \texttt{domain}, o) \rightarrow (p, \texttt{domain}, o)$
ext4	$(p, \texttt{subPropertyOf}, p_1)$, $(p_1, \texttt{range}, o) \rightarrow (p, \texttt{range}, o)$

Table 2.2 – Subset of RDF entailment rules typically used in data management

A fact is either a *class assertion* stating that some identifier has some *type*, or a *property assertion* stating a *relation* between two identifiers or between some identifier and some constant. For instance, a KB may state that i_1 is a person, whose name is Steve Jobs and who is employed by i_2 with the following triples:

$$(i_1, \texttt{type}, \text{Person}), (i_1, \text{name}, \text{"Steve Jobs"}), (i_1, \text{emplBy}, i_2).$$

On the other hand, domain knowledge is expressed by establishing so-called *ontological constraints* between the classes (types) and properties (relations) allowed to state facts. Such constraints are *subclass and subproperty relationships*, a.k.a. ISA constraints, as well *typing constraints for property attributes*: the first attribute is called *domain* while the second is called *range*. For example, the above KG may additionally state that employees are persons, only employees are employed, only organizations are employers and that

being employed by is a particular case of working for:

(Employee, `subClassOf`, Person), (emplBy, `domain`, Employee),
(emplBy, `range`, Organization), (emplBy, `subPropertyOf`, workFor).

Importantly, RDF KBs model both explicit and implicit data and knowledge triples. The explicit ones are those stored in the KB, e.g., the triples of the above example, while the implicit ones are those that can be derived from the explicit ones and *entailment rules* —inference rules— from the RDF standard [73]. The RDF entailment rules typically used in data management are gathered in Tab. 2.2, e.g., [10]. They provide the deductive aspect of RDF and allow to compute the implicit triples that can be deduced by looking at data and knowledge triples in the KB. For instance, above, from the fact i_1 *is employed by* i_2 and the constraint *only employees are employed*, the implicit fact i_1 *is an employee* is inferred using the rule rdfs2 in Tab. 2.2, i.e., the implicit triple (i_1, `type`, Employee) holds in the KB although it is not explicitly stored there. Also, from the two constraints *only employees are employed* and *employees are persons*, the implicit constraint *only person are employed* is inferred based on the rule ext1 in Tab. 2.2, i.e., the implicit triple (emplBy, `domain`, Person) also holds in the KB.

In this thesis, we exploit the *fine-grained semantic description of entities* that RDF KBs make available in terms of their types—e.g., the entity i_1 representing Steve Jobs is known as a Person and implicitly as an Employee in the KB —and of their relationships— e.g., i_1 is known to be employed by i_2 and implicitly to work for i_2. Crucially, to fully take advantage of the explicit *and* implicit triples of a KB, in particular for the entity linking technique, we assume that the KB is *saturated*, i.e., *all* its implicit triples have been computed and explicitly added to the KB. KB saturation, a.k.a. *closure*, is a reasoning step defined in the RDF standard [73] that many RDF data management tools implement, like the widely used Apache Jena platform [5]. In case of KBs that are not saturated, we perform the saturation using entailment rules from RDF standard [73] and their implementation within Apache Jena platform. The saturation step allows to explicitly add implicit triples that are not directly stored in the RDF KB, thus making full advantage of entities —explicit and implicit— interrelationships in RDF KBs.

5. `https://jena.apache.org`

2.2 Word Embeddings

Word embeddings allow encoding words as points in a multidimensional space. There exist several embedding models such as [50, 57, 5, 58, 18]; their basic intention is that semantically similar words end up in near coordinates within the target multidimensional space. Thus the cosine similarity between two points in the multidimensional space at hand reflects their semantic proximity. Word embedding are based on the definition of a sliding window as local context in [50, 57, 5], global word co-occurrences are further incorporated in [57] to model both local and global word interactions, while sub-words as n-grams are used instead of tokens in [5]. Finally [58] defines word embedding based on their positions in a given sentence, meaning that the representation of a word is changing depending on the sentence it is used in. These embedding models have similar performance for the task of entity linking. Therefore in this work, we consider the skip-gram embedding model [50] which relies on the famous formula of Firth [25]: "You should know a word by the company it keeps". For skip-gram defined in [50], word embedding are learned to predict for a given word, the words that appear in its context, i.e., surrounding words, based on dot products between vector representations of words. More formally, given a large training corpus represented as a sequence of words $w_1, ..., w_N$, the objective of the skip-gram model is to maximize

$$\sum_{t=1}^{N} \sum_{c \in C_t} \log P(w_c|w_t) \tag{2.1}$$

where the context C_t is the set of indices of words surrounding word w_t in a sliding window. A usual choice to define the probability of a context word using dot product is the softmax function

$$P(w_c|w_t) = \frac{\exp(v_c^T v_t)}{\sum_{i=1}^{N} \exp(v_i^T v_t)} \ , \tag{2.2}$$

where v_i denotes the vector representation of word w_i. These vectors can efficiently embed word semantics, especially when the model is trained using a huge dataset such as Wikipedia or Google News.

Word embedding and cosine similarity are typically used as ingredients for entity linking approaches to link entity mentions in text to entities in KBs.

2.3 Knowledge Bases Embeddings

KB embedding techniques represent entities as points in a multidimensional space and the relations as transformations between those entities. There exist several embedding models such as [7, 45]; their basic intention is to embed the semantics of the KB into a multidimensional space. Thus the cosine similarity between two points (embeddings of the entities) reflects their semantic distance to some extent. Most approaches derive from the TransE [7] KB embedding technique. Given a KB represented by a set of triples (h, r, t)— indicating a relation r between a head entity h and a tail one t —TransE learns vector representations for entities and relations such that $v_h + v_r \approx v_t$, if the triple (h, r, t) holds, where v_x denotes the embedding of a given element x. The resulting embedded space obviously bears the semantics of the KB, where each learned translation in the embedded space correspond to a relation in the KB. Formally, the KB embedding objective function, which takes care of the semantics of the KBs, is defined in the case of TransE [7] as

$$L = \sum_{(h,r,t)\in T^+} \sum_{(h',r,t')\in T^-} [\lambda + f((h,r,t)) - f((h',r,t'))]_+ \ , \tag{2.3}$$

where $f()$ is a triple scoring function, here $f((h,r,t)) = ||v_h + v_r - v_t||_2^2$, $\lambda > 0$ is a margin hyperparameter, and $[x]_+ = max(x, 0)$. T^+ and T^- denote the sets of positive and negative triples respectively. The latter, T^-, is obtained by replacing the head or the tail of an existing triple with another non-related entity, while the positive triples are explicitly represented in the KB. In plain words, L enforces the TransE translation property in the embedded space, minimizing the cost on positive triples—those bearing an actual relation—and maximizing on negative triples.

KB embedding techniques allow learning vectors that can embed efficiently entities semantics, and allow defining a semantic distance between entities within the multidimensional space, e.g., cosine similarity. The interest of this semantic distance is mostly twofold: (a) it can be used as an entity relatedness measure for entity linking; (b) it is instrumental for entity alignment as it allows computing the similarity between two entities in different KBs with the ultimate goal to match them in the context of entity alignment.

2.4 Jointly Embedding Words and KBs

Learning a joint representation of words and KBs, e.g., [53, 78, 31] allows encoding in the same embedding space words and entities as points in a multidimensional space where the distance between mentions in text and entities in a KB is semantically correct, as we cannot interpret the distance between two multidimensional points that are not in the same multidimensional space. The joint embedding technique EAT [53] learns representations for words and entities similarly to skip-gram. The basic idea of EAT is to extend words contexts to the entities they are referring to in a text. Formally if we have m as a text (anchor) of an hyperlink referring to an entity e, first m is redefined as the couple (m, e); then the pair (m, e) is distributed to the context of m at training time; finally the previously defined skip-gram model is used to learn the joint representation. EAT allows to learn entities representation based on the contexts of their mentions in a given text corpus. Anchors are differently used in [78] to jointly learn words and entities embeddings in the same multidimensional space. Three different models are jointly learned, namely a word embedding using skip-gram, a KB embedding model and an anchor context model. The KB embedding model is similar to skip-gram and is trained to predict the links associated with an entity in a KB. The anchor context model allow bridging word embedding using skip-gram and entity embeddings using the KB embedding model. The underlying idea is to use the anchors of the hyperlinks referring to an entity as its context, and then train a model to predict these anchors for a target entity. The KB embedding model and anchor context model use an objective function similar to Eq. 2.1, while only changing the definition of the context C_t. For the KB embedding model, the context of an entity is its linked entities in the KB, while the words referring to an entity are used in the anchor context model to define its context. The joint embedding model in [78] can reflect only Wikipedia properties as it is mainly based on the hyperlinks structure of Wikipedia (anchors). Therefore, it is only supported by Wikipedia-like KBs. Finally [31] jointly embeds entities and mentions in the same space leveraging mention context (anchors), entities description and fine-grained types (Person, Politician, Governor, Organization, Location, Country, City, etc.). Formally, mention-entity similarity is defined for a mention m that refers to an entity e as:

$$P_{text}(e|m) = \frac{exp(v_m, v_e)}{\sum_{e' \in C_m} exp(v_m, v_{e'})} \tag{2.4}$$

where v_x is the embedding of the element x in the joint embedding space and C_m is the candidate entities of the mention m. Moreover, a CNN is used to encode entity description as a fixed size vector. The probability that the description of an entity e is being encoded by a vector v_{desc} is defined similarly to 2.4. The last model which encodes fine-grained types, learns an encoding for entity *types* from a KB so that entities get projected close to their associated types. The joint representation is learned by minimizing the log-likelihood of the three models at once.

In general, anchor-based joint embedding models make use of a mapping between words and entities. However this mapping may not be available, especially in the case of an ad hoc KB constructed for a particular need. Therefore we advocate for entity relatedness measures instead of joint embedding techniques.

2.5 Conclusion

This chapter gives an overview about certain aspects of text and KBs representations which we discuss in this thesis. We first describe how the RDF data model encodes KB semantic. We also show how words and entities —that are different— can be embedded into a multidimensional space (separately or jointly). Embedding techniques enable not only to define a semantic distance between words in text and entities in a KB, but also to define semantic distance between entities being in the same KB —entity relatedness measure— or in two different KBs –entity alignment—. Semantic entity relatedness measures are crucial ingredients of the collective entity linking. Hence, in the first part of this thesis we propose different entity relatedness measures that use the semantics of RDF KBs to define semantic distances between entities, which we use to improve the collective entity linking. Particularly, in the next chapter, we study the effectiveness of using semantic entity relatedness measures that account for the number of direct relations between the entities of the RDF KB at hand, for the collective entity linking.

Chapter 3

Using Knowledge Base Semantics in Context-Aware Entity Linking

Entity linking as mentioned previously is a core task in textual document processing, that consists in retrieving the entities of a KB that are mentioned in a text. Entity linking facilitates a variety of tasks such as semantic search [4] and information extraction [37, 38, 39].

Standard entity linking systems usually implement three well-established steps [63] to solve mention ambiguity in text. Named entity recognition (NER) is first performed to identify the entity mentions in a document. Candidate entities from the KB under consideration are then generated for each mention found. Finally, every mention is linked to one of its candidate entities in a so-called linking step. For this last step, two types of approaches can be found in the literature depending on whether the linking is performed *independently* for each individual mention, e.g., [49, 26, 31], or *collectively* for all mentions at once, e.g., [34, 28, 78, 13, 59, 47, 41]. In the first case, called *entity-by-entity linking*, entity linking approaches rely on the independence hypothesis meaning every mention in a text is assumed to be independent from other mentions. This hypothesis suggests to link a mention in text to a candidate entity on the sole basis of some similarity between the mention and the candidate entities, a.k.a. *local* score. Nonetheless, mentions in text are tied to a particular context within a coherent document, they are somehow semantically related, i.e., *mention-to-entity linking decisions are interdependent*. Therefore, in the case of *collective* entity linking, the local score is complemented with a *global* score reflecting to which extent the candidate entities chosen for the mentions under consideration are related in the KB. Thereupon, collective entity linking makes a fine-grained use of entities interrelationships, according to an *entity relatedness measure*. The latter is used to devise the global score that alleviates the independence hypothesis. Hence, entity relatedness measures allow quantifying the coherence of entity linking decisions based on entities interrelationships in the reference KB. Hence, we propose to devise a semantic entity

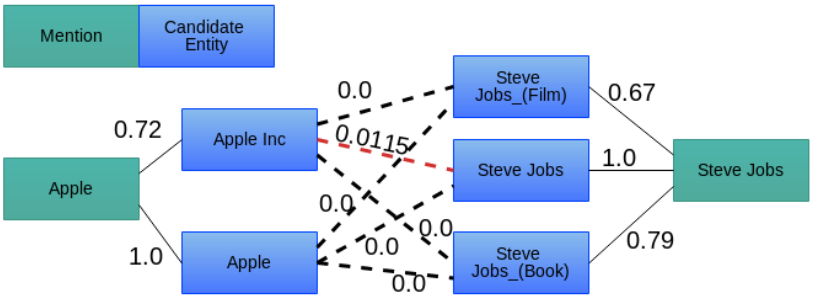


Figure 3.1 – Illustration of a mention-entity graph within a document: the two mentions (Apple and Steve Jobs) are linked to candidate entities with some local score (solid lines) while the entity relatedness appears as weighted dotted lines. In this toy example, the strong relation between the entities Steve Jobs and Apple Inc helps in jointly selecting those two.

relatedness measure, that reflects entities interrelationships, accounting for entity relations in an RDF KBs to the possible extent. The work we present in this chapter, has been published in [24].

In the remaining of this chapter, we discuss the definition of a lightweight collective entity linking system that benefits to the extent possible from entities interrelationships in an RDF KB. First, Sec. 3.1 discusses the state of the art, introducing concepts and notations and allowing to highlight how our proposed solution for the linking differs from the collective entity linking in the literature. Then, we describe our method in Sec. 3.2 showing in particular how it efficiently incorporates entity interrelationships in the linking process, accounting for a novel fine-grained *weighted semantic relatedness measure* (WSRM). Finally, we extensively evaluate our method on popular entity linking datasets and compare it to state-of-the-art approaches in Sec. 3.3 before discussing the perspectives opened by our new entity relatedness measure in Sec. 3.4.

3.1 Related Work

Entity linking (EL) has been widely investigated in the literature, as reported in the recent survey [63]. In this thesis, we mostly restrict the discussion to the ranking of candidate entities for mentions, since this is the step that mainly concerns our contributions. We focus on entity-by-entity linking in Sec. 3.1.1 before considering collective linking in Sec. 3.1.2. Limits of these approaches are discussed in Sec. 3.1.3.

3.1.1 Entity-By-Entity Linking

Assuming the mentions in a text to be independent one from another, entity-by-entity methods are based on the similarity between some entity mention and its candidate entities in a KB. Basically, they search the candidate entity that maximizes a so-called local similarity measure between a mention and its candidate entities. In Fig. 3.1 which depicts mentions and candidate entities within a document, only the relationship between a mention and its entity are considered.

This can be formalized as

$$\hat{e} = \arg\max_{e_i} \phi(m, e_i) \tag{3.1}$$

where e_i is a candidate entity, m is an entity mention, and ϕ is the local score function.

Entity-by-entity linking is studied in several publications [49, 16, 19, 20, 26, 31]. Words and entities are embedded in vectors in [16, 19, 49] and a cosine similarity is used to compute the local score (see Chapter 2), while [20] and [26] use *Wikipedia popularity*. This popularity, a.k.a. *commonness*,

$$\text{pop}(m, e) = \frac{n(m, e)}{\sum_{e' \in W} n(m, e')} \tag{3.2}$$

can be defined as the probability that a mention m is used as the text (anchor) of a hyperlink referring to an entity e, with W the set of all Wikipedia pages and $n(m, e)$ the number of times m occurs as an anchor for e in some Wikipedia page. [26] further combines popularity with a convolution neural network (CNN) to extract topic vectors from both the context of the mention (words surrounding it) and the context of an entity (its Wikipedia web page). Popularity and CNN's features are fed into a final logistic regression to perform the linking. Finally [31] jointly embeds entities and mentions in the same space leveraging their model described in Sec 2.4 and defines the local score as:

$$\phi(m, e) = pop(m, e) + P_{text}(m|e) - pop(m, e) * P_{text}(m|e) \tag{3.3}$$

where $P_{text}(m|e)$ is defined in Eq. 2.4. One can notice that Eq. 3.3 is similar to modeling the disjonction of the two scores $pop(m, e)$ and $P_{text}(m|e)$, therefore this equation unifies in the same local score *pop* and P_{text}

Entity-by-entity approaches consider mainly mention-entity score, ignoring entities

interrelationships in the reference KB.

3.1.2 Collective Entity Linking

Collective entity linking (CEL) improves on entity-by-entity linking by not only considering the similarity between an individual mention and its candidate entities, but also taking into account the intricate interrelationships that candidate entities of the different mentions may have. To this aim, the local score function $\phi()$ is complemented with a function $\psi()$ reflecting *entity relatedness*, i.e., affinity between entities. The CEL problem can be thus formalized as

$$(\hat{e}_1, ..., \hat{e}_n) = \underset{e_{1,...,n}}{\arg\max} \left(\sum_{i=1}^{n} \phi(m_i, e_i) + \sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} \psi(e_i, e_j) \right) \tag{3.4}$$

where n is the total number of mentions in a text. The quantity $\sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} \psi(e_i, e_j)$ is called *global* score, and measures the coherence of the linking decision for the mentions in a document. Maximizing the global score allows to define the best matching for all the mentions in a document at once based on an entity relatedness measure $\psi(e_i, e_j)$.

Popular entity relatedness measures in the literature are the cosine similarity using entity embeddings (see Chapter 2), the Wikipedia hyperlink-based measure (WLM) [52]

$$WLM(e_i, e_j) = 1 - \frac{\log\left(\dfrac{\max(|\mathrm{IN}_{e_i}|, |\mathrm{IN}_{e_j}|)}{|\mathrm{IN}_{e_i} \cap \mathrm{IN}_{e_j}|}\right)}{\log\left(\dfrac{|W|}{\min(|\mathrm{IN}_{e_i}|, |\mathrm{IN}_{e_j}|)}\right)} \ , \tag{3.5}$$

the Normalized Jaccard similarity (NJS) [59]

$$\mathrm{NJS}(e_i, e_j) = \frac{\log\Big(|\mathrm{IN}_{e_i}| \cap |\mathrm{IN}_{e_j}| + 1\Big)}{\log\Big(|\mathrm{IN}_{e_i} \cup \mathrm{IN}_{e_j}| + 1\Big)} \ , \tag{3.6}$$

and the reference binary indicator (Ref) [1, 46]

$$\mathrm{Ref}(e_i, e_j) = \begin{cases} 1 & \exists r, (e_i, r, e_j) \in KB \\ 0 & \text{otherwise} \end{cases} \tag{3.7}$$

In these equations, IN_{e_i} is the set of incoming hyperlinks in the Wikipedia page for e_i,

and r is a relationship between e_i and e_j in a broad sense (a hyperlink for the Wikipedia KB or a relation for an RDF KB). For example, *WLM* is used in [52, 34, 46], *Ref* in [1, 46], cosine similarity between entities in [53, 78, 48, 47], while a combination of the three is used in [59].

Beyond the entity relatedness measure, collective linking techniques also differ on their linking strategy, either relying on an approximate optimization of Eq. 3.4 [53, 46, 78, 41, 47] or on a *mention-entity graph* [34, 52, 28, 43, 48, 1, 59, 13, 75] such as the one illustrated in Fig. 3.1.

Optimization-based techniques aim at computing the solution to Eq. 3.4, which is known to be NP-hard [3]. They therefore compute an approximation of the solution. [53, 78] jointly embed words and entities in the same space and train a classifier with local and global scores to select the best solution. Thus, they replace the global optimization with local classification accounting for features depicting local and global scores. [47] reduces the problem to a sub-matrix search and solves the optimization of Eq. 3.4 using a gradient descent. [46] first selects for each mention, the best candidate entity based on an entity type scheme like Eq. 3.2, then solves the optimization of Eq. 3.4 by using only the previous selected candidate entities in the computation of global scores. Finally, [41] advocates for models that jointly solve named entity recognition and entity linking using only word and character embedding [27].

Mention-entity graph-based techniques build a graph whose nodes are entity mentions and candidate entities, and whose edges connect either some mention to one of its candidate entities or candidate entities of different mentions. Edges are weighted, in the first case with the local score, in the last case with an entity relatedness measure [34, 48]. A slightly different graph is used in [1]: nodes denote (mention, candidate entity) pairs and edges connect nodes with non-zero score using the entity relatedness measure Ref. Also, [75] constructs a graph leveraging mentions coreferences and tries to match it with a candidate entity graph. [13] used a graph convolution network (GCN) to perform the linking using local and global scores. [28, 43] use a factor graph based on *loopy belief propagation* (LBP), a.k.a. sum-product message passing algorithm which allows inference over a graph. Lastly [59] proposes to solve the CEL with analogy to the minimum spanning tree problem using cosine similarity, WLM and NJS.

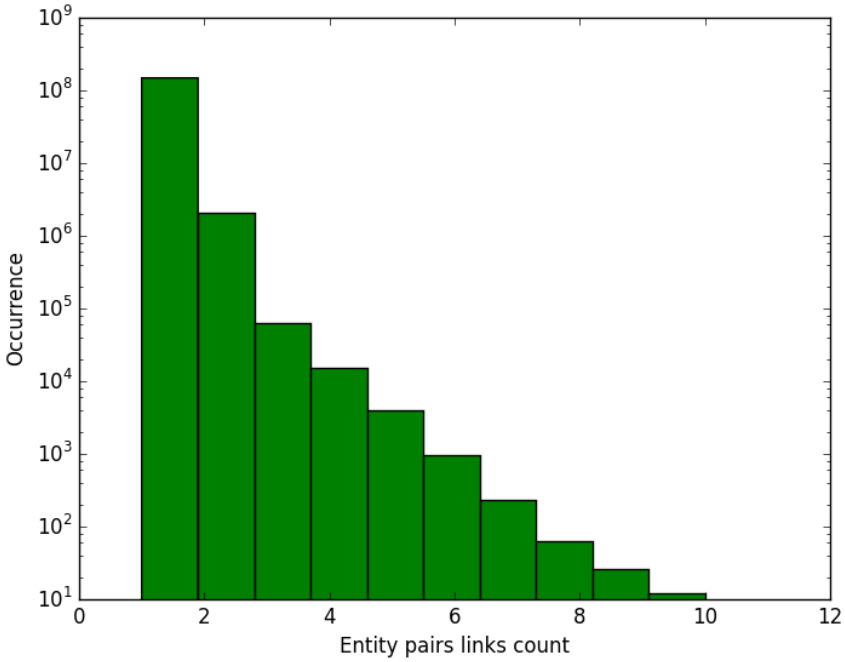


Figure 3.2 – Distribution of entity pairs (Y-axis) per number of relations between entities (X-axis) in the BaseKB RDF KB.

3.1.3 Limits of Entity Linking Techniques

In this thesis, we argue that collective entity linking, which already improves on entity-by-entity linking, can be further improved by capitalizing on the precise semantics of the relationships between entities that a structured RDF KB provides. In particular, entity relatedness measures used so far in CEL techniques either build on the loose semantics of connections between entities in the Wikipedia hyperlink graph (e.g., WLM, NJS or Ref), which only indicates that connected entities are *somehow* related, or simply on the *existence* of some relation between entities in an RDF KB (e.g., Ref). When a RDF KB is used, we advocate that going beyond the simple existence of some relation between two entities, by taking into account all the relations that are connecting these entities, is feasible and beneficial. To give an insight on the valuable extra information this might yield, Fig. 3.2 shows the distribution of entity pairs as a function of the number of relations between two entities in the BaseKB RDF KB. For instance, there are approximately 10^8 pairs of entities connected by a single relation, and about 10 pairs connected by 10 relations. It is worth noting that all these pairs of entities are indistinguishable if we

only consider the existence of a relation between them as *Ref* does. We incorporate this information into a new relatedness measure to model that the more connected entities are in the KB, the more likely they are mentioned together in some text document. We then follow a lightweight supervised CEL technique, whose global scores use this measure, and which does not require constructing the mention-entity graph.

3.2 Collective Linking with KB Semantics

Our main contribution to the collective linking task lies in the definition of a novel entity relatedness measure, designated as weighted semantic relatedness measure (WSRM), that takes advantage of the KB structure in the candidate selection stage of the standard EL pipeline. This measure is combined with local mention-entity similarity measures in a classification step whose goal is to predict whether a link should be made or not between a mention and a candidate entity. The classifier operates independently on each mention-entity pair, however taking as input features depicting the local and global scores. Taking into account the context in which a mention-entity pair appears enables a form of collective linking at a low computational cost, replacing global optimization with an ensemble of local classifications.

Entity linking is composed of three separated stages as explained before. Following many previous studies, e.g., [78, 28, 53, 59, 13], we disregard the named entity mention detection stage, for which a number of online accurate tools are available [1], [2] and assume that we know for each document the mentions and their type (person, organization, location, etc.). The motivation behind this choice is to directly evaluate our linking stage without the errors coming from the NER. For the candidate generation stage, a standard architecture is chosen as in [63], relying on Wikipedia for practical reasons. The candidate selection stage accounts for the WSRM between candidate entities within the document in an efficient manner, relying here on BaseKB, one of the various RDF KB providing structured knowledge about Wikipedia entities. Note that while BaseKB is used here, we are conceptually not limited to this KB as the WSRM can account for any RDF KB. The choice of BaseKB was mostly driven by the fact that various baselines use the very same KB and that candidate generation relies at this stage on Wikipedia entities, thus requiring a mapping of the reference KB to those entities.

1. https://nlp.stanford.edu/software/CRF-NER.html
2. https://github.com/mit-nlp/MITIE

In the remainder of this section, we represent a document D by its set of entity mentions, $D = (m_1, ..., m_n)$ where n is the number of mentions. For each mention m_i, $C(m_i) = (e_{i1}, ..., e_{ik})$ denotes the set of its candidate entities. We first present the new entity relatedness measure WSRM, as it constitutes the core contribution of this chapter, then the presentation follows the logical steps of the whole process, namely candidate generation, local score definition, and the classification-based actual linking based on WSRM.

3.2.1 Entities Relatedness

As highlighted in Sec. 3.1.2, a wide range of entity relatedness measures (a.k.a. coherence) has been defined and used in the literature to establish a score that reflects the relationship between two entities in a KB. But only a few reflect a clear semantic similarity measure as defined by the interrelationships entities have in an RDF KB. One notable exception is the $\text{Ref}(e_i, e_j)$ measure of Eq. 3.7 which solely provides a binary relatedness measure.

To take full advantage of the KB semantics, we introduce a weighted semantic relatedness measure (WSRM) based on the total number of relations two entities share in the KB. The key idea in this measure is to not only express the existence of relations in the KB between the entities at hand, but also weight the relation between entities, where the more relations between the entities, the stronger their relationship. Formally, we define the relatedness between two entities e_i and e_j as

$$\text{WSRM}(e_i, e_j) = \frac{|\{r \mid (e_i, r, e_j) \in \text{KB}\}|}{\sum_{e' \in E} |\{r' \mid (e_i, r', e') \in KB\}|} \ , \tag{3.8}$$

where E denotes the set of entities in the KB and $|S|$ the cardinality of the set S. We assume the KB to be saturated (all implicit triples are made explicit) and we disregard triples corresponding to class assertions and RDF knowledge (e.g., type, subPropertyOf, domain, range, sameAs) (see Chapter 2).

Because the directions of the relations are somewhat arbitrary in KBs, depending on how the relation vocabulary was designed (e.g., think about the *publishes* and *publishedBy* symmetric RDF properties), we use a symmetric version of WSRM defined as

$$\psi(e_i, e_j) = \frac{1}{2} \left(\text{WSRM}(e_i, e_j) + \text{WSRM}(e_j, e_i)\right) \ . \tag{3.9}$$

Conceptually, WSRM(e_i, e_j) is similar to popularity however applied to two entities rather than to a mention and an entity: it exploits the number of relations two entities e_i and e_j share in the KB. As we normalize over the KB, this measure gives a probability that e_i is related to e_j: a value of 1 means e_i is connected only to e_j in the KB, while a value of 0 means the absence of connection in the KB. The higher the number of relations between two entities, the higher their probability to be mentioned together in a text. Thus selecting candidate entities that have a large number of relations in the KB is intuitively bound to maximize the CEL objective function of Eq. 3.4. Note finally that while Eq. 3.8 only considers direct relations between the two entities, i.e., there is an RDF triple linking e_i and e_j one way or another, it opens the door to fully exploit RDF KB semantics for the collective entity linking.

3.2.2 Candidate Entities Generation

To generate candidate entities from the KB for each mention in a document, we chose a simple yet efficient method exploiting Cross-Wiki [65]. Cross-Wiki is a dictionary computed from a Google crawl of the web that stores the frequency with which a mention links to a particular entity in Wikipedia. We used the same Cross-Wiki dictionary as in [31] [3]. Each entry of the dictionary corresponds to a possible entity mention and provides a list of Wikipedia pages (i.e., candidate entities) along with their associated popularity scores.

This list is directly used for candidate generation whenever a mention appears in the dictionary. The dictionary entries are normalized by removing all punctuation marks and converting to lowercases. For example, the generation of candidate entities for "Steve Jobs" with Cross-Wiki leads to

cross-wiki["stevejobs"] => [['7412236', 0.99],['5042765',0.01]]

where '7412236' is the id of the Wikipedia page "Steve Jobs", with a score of 0.99, and '5042765' is the id of the Wikipedia page "God" (sic!). For mentions absent from cross-wiki, we perform a request on Wikipedia using the text of the mention, and collect the resulting Wikipedia pages as the candidate entities.

3. https://drive.google.com/uc?id=0Bz-t37BfgoTuSEtXOTI1SEF3VnM&export=download

3.2.3 Local Mention-Entity Score

The local relevance score $\phi(m_j, e_{ij})$ between a mention m_i and a candidate entity e_{ij} can be established between the mention and the candidate entity name only (name of the corresponding Wikipedia page), but can also consider the context (e.g., the surrounding words of the mention, and the description of the entity respectively). In this work, we consider two local score functions, namely the cosine similarity between m_i and the Wikipedia title of e_{ij} in an embedded space, and the popularity as defined in Eq. 3.2. The former reflects the geometric (and thus semantic) proximity in the embedded space of one mention and one entity. The latter is known to be a good estimation of the similarity between a mention and an entity.

3.2.4 Supervised Collective Entity Linking

The last step is to decide which candidate should be retained for each mention within the document. To do so in collective linking and exploit the WSRM entity relatedness measure, any of the solutions mentioned in Sec. 3.1.2 could be used, exploiting the mention-entity graph one way or another. These approaches are however computationally heavy and their cost grows rapidly as the number of mentions and candidates increases. We thus adopt a supervised approach similar to [53, 78], where a binary classifier is trained to predict whether a mention and a candidate entity are related (1) or not (0). However, in the collective linking setting, the classifier relies on features related to the mention-entity pair as well as on contextual (global) features accounting for the relatedness of the entity with candidates from other entity mentions in the document.

For practical reasons, it is better to have contextual features of fixed size, which for a candidate entity e_{ij} aggregates the entity relatedness scores $\psi(e_{ij}, e)$ for all $e \in C(m_l), l \neq i$. The alternative would be to consider all scores up to a maximum, zeroing non existing scores, but aggregation appears much more simple and is experimentally shown efficient (see Sec. 3.3). We consider a conjunction of simple aggregators such as the sum:

$$\mathrm{S}(e_{ij}; D) = \sum_{l=1, l\neq i}^{n} \sum_{e \in C(m_l)} \psi(e_{ij}, e) \ , \tag{3.10}$$

mimicking the global term in Eq. 3.4, or the k maximum values:

$$\mathrm{M}^k(e_{ij}; D) = \max_{l=1, l\neq i}^{n} @\mathrm{k} \max_{e' \in C(m_l)} \psi(e_i, e) \tag{3.11}$$

where max @k is the k^{th} highest value. Note that combining aggregators is beneficial: on the one hand, using the sole sum can indeed introduce noise since not all candidate entities are relevant for the linking; on the other hand, leveraging the sole max can be very drastic because only candidate entities with maximum relatedness score are kept. But, as evidenced in [59], entities mentioned in a document are not necessarily the most connected in the KB. Hence we retained the S, M_1, M_2 and M_3 as global contextual features, which can be seen as a kind of flexibility in selecting and aggregating the relatedness scores.

Finally, a binary logistic regression classifier is trained to predict whether a mention and an entity match or not, taking as input features the two mention-entity scores (cosine similarity and popularity) and the 4 global contextual features. At linking time, this classifier is used independently for all mentions m_i, considering all pairs (m_i, e) with $e \in C(m_i)$ and retaining the best one as the entity corresponding to m_i, i.e., e_{ij} with $j = \arg\max_k \text{logreg}(m_i, e_{ik})$ where logreg() is the binary logistic regression classifier.

3.3 Experiments

This section investigates the benefit and the accuracy of both our novel entity relatedness measure WSRM, and our lightweight collective linking technique that builds upon it. We describe our experimental setup in Sec. 5.3.1. Then Sec. 3.3.2 proposes an ablation study of our method in order to demonstrate the impact of local and global scores. Finally Sec. 5.3.2 compares our CEL approach to state-of-the-art entity-by-entity and CEL competitors.

3.3.1 Experimental Setup

Knowledge Base

We make use of Wikipedia in the candidate generation stage and of BaseKB [4], an RDF knowledge base derived from Freebase which contains over one billion facts (i.e., triples) about more than 40 millions subjects, for semantic relatedness measure. As previously mentioned, the crucial interest of such an RDF KB over Wikipedia resides in the fact that both its entities and their interrelationships bear a precise semantics.

A mapping between Wikipedia and BaseKB entities is used to enable taking advantage of Wikipedia in the early stages of the process, in particular the fact that names in

4. http://basekb.com/

Dataset	Nb. docs	Nb. mentions	Avg nb. mentions/doc
TAC-KBP 2016 eval	169	9231	54.6
TAC-KBP 2017 eval	167	6915	41.4
AIDA-train	846	18519	21.9
AIDA-valid	216	4784	22.1
AIDA-test	231	4479	19.4
Reuters128	128	881	6.9
RSS-500	500	1000	2

Table 3.1 – Statistics on the datasets used.

Wikipedia are meaningful unique identifiers, and of BaseKB semantics in the linking stages. To ensure consistency, we discarded from BaseKB entities with no corresponding Wikipedia entry, resulting in approximately 4M entities in the RDF KB.

Datasets

Experimental results are reported on four standard datasets with different characteristics, including both short and long documents as well as formal (news) and informal (forum) texts:

— CoNLL-AIDA is an entity annotated corpus of Reuters news documents introduced by Hoffart et al. [34]. It is much larger than most of the other existing EL datasets, making it an excellent evaluation target. Data is divided into three parts: Train, AIDA-A (used for validation) and AIDA-B (used for evaluation). The original target KB for CoNLL-AIDA was YAGO but a recent update allows linking to BaseKB.

— TAC-KBP Entity Discovery and Linking (EDL) 2016-2017 datasets are newswire and forum-discussion documents originally collected for the TAC Knowledge Base Population Entity Discovery and Linking 2016 and 2017 international evaluation campaigns [39]. We only used the gold-standard where entity mentions are already annotated.

— Reuters128 [62] is a small dataset that contains 128 economic news articles taken from the Reuters-21587 corpus.

— RSS500 [62] contains 500 documents created from RSS feeds including all major worldwide newspapers and a wide range of topics, e.g., World, U.S., Business, Science, etc.

Table 3.1 gathers key figures and statistics for each of the datasets. For experiments on

the Reuters128 and RSS-500 corpora, training of the classifier is performed on the AIDA train dataset due to the small size of these two datasets.

Practical details

Word embedding of dimension 100 trained on a dump of Wikipedia from October 2018 are used for all the experiments. Only words appearing at least 5 times are retained in the embedding.

At classification time, for candidate entities generated with Cross-Wiki that are absent from Wikipedia, all the features were set to 0 as, apart from the popularity provided by Cross-Wiki, features for out-of-KB entities cannot be computed. We also limit classification to entity mentions that have a corresponding entity in the KB: in other words, we disregard mentions for which the ground truth points to entities not present in BaseKB. Note also that for efficiency reasons, the WSRM entity relatedness measure was precomputed for all pairs of entities in BaseKB, thus limiting the computational cost at linking time.

Evaluation is provided in terms of F1 score, where precision $P = \frac{|G \cap S|}{|S|}$ and recall $R = \frac{|G \cap S|}{|G|}$ are calculated between the linking in the gold-standard (G) and the linking given by a system (S).

3.3.2 Ablation Study

As highlighted in Sec. 3.1, CEL is supposed to improve over entity-by-entity linking by taking into account the interrelationships between the candidate entities of the different mentions. Considering the two mention-entity scores and the four contextual features respectively retained and proposed in Sec. 3.2.4, the ablation study described here aims at demonstrating the benefit of considering together local and global scores in our CEL technique, also attesting again the relevance of our weighted entity relatedness measure.

Using TAC-KBP 2016 as training data, several versions of our classifier were thus trained, which consider either local scores only (i.e., an entity-by-entity linking version of our approach) or also some or all the global features that we proposed. Table 3.2 presents the performance of those different versions on TAC-KBP 2017 in terms of linking accuracy. [5]

5. Note that the results reported in Table 3.2 for our approach are not directly comparable to those obtained by TAC KBP EDL 2017 systems [39] since we do not use an end-to-end system and omit the entity recognition stage.

Features	**F1 score**
$popularity$	0.723
$popularity + cosine$	0.729
$popularity + cosine + S$	0.732
$popularity + cosine + S + M_{1,2,3}$	0.750

Table 3.2 – Linking accuracy (F1 score) on the TAC KBP-2017 dataset. Popularity and cosine similarity are the local mention-entity scores; S and $M_{1,2,3}$, the global features, are defined in Eq. 3.10 and Eq. 3.11 respectively.

Combining all local and global features leads up to the best result (0.75). This can be further improved by using entity type information, filtering candidate entities by type. Given entity mentions in the documents together with their types (person, location, organization, etc.), we only retain candidate entities for which the BaseKB type corresponds. Taking into account type information, we achieved a F1 score of 0.79.

3.3.3 Comparison to State-of-the-Art Approaches

Finally, we compare our CEL approach to a series of EL systems which report state-of-the-art results on the AIDA-A, AIDA-B, Reuters128 and RSS500 datasets, namely:

— a CNN approach to capturing semantic similarity for entity linking [26] where one CNN is used to extract contextual features (CNN);

— an end-to-end neural attention-based entity linking [41] where entity recognition and linking are jointly solved using words and characters embedding (End-to-End);

— a graph-based approaches operating on the mention-entity graph, resp. AIDA [34], AGDISTIS [69] and Babelfy [54];

— a neural collective entity linking where a graph convolution network is used to label mentions with entities from the KB [13]. Cosine similarity between entities is used as the relatedness measure (NCEL);

— a probabilistic bag-of-hyperlinks model for entity linking which uses counts of co-occurrences of entities along with a loopy belief propagation algorithm [28] (PBoH);

— VINCULUM [46] where both Ref (Eq. 3.7) and WLM (Eq. 3.5) are used as relatedness measure and a two-step approach is used for collective linking.

The list of EL systems above can be split into entity-by-entity approaches (CNN, End-to-End) and collective ones (AIDA, PBoH, AGDISTIS, Babelfy, VINCULUM, NCEL). For fair comparison, we point out that [26] does not use global score and both End-to-End

and VINCULUM use an automatic entity recognition stage, where we assume we have a perfect one. The scores for AIDA, PBoH, AGDISTIS and Babelfy were obtained from the online available platform GERBIL [70] and scores for [26, 13, 46, 41] are taken from the original papers, those methods being absent from the platform.

Results of the entity linking process evaluated in terms of micro-averaged F1 classification scores are reported in Tab. 3.3 except for VINCULUM which reports a different metric. On all four datasets, the proposed WSRM with the logistic regression classifier does outperform the AIDA, PBoH and NCEL collective linking approaches by a large margin (see below for statistical significance). The method is also competitive with respect to the end-to-end approaches. One interesting point to note is that AIDA and PBoH perform very differently on Reuters128 and RSS-500. This can be explained by the low density of mentions in the RSS-500 dataset with an average of 2 mentions per document. In these conditions, optimization-based approaches like PBoH do not perform well on short text unlike graph-based approaches. On the contrary, classification seems to be little affected by those drastic statistics.

Comparison with VINCULUM is reported in Tab. 3.4, using the macro averaged F1 classification as reported in [46]. While not directly comparable because VINCULUM does rely on automatic named entity recognition, the difference in macro averaged F1 scores of approx. 17 points is unlikely to be solely explained by entity recognition errors. As most CEL methods do use the same local scores, i.e., popularity and cosine similarity based on the skip-gram model, we can conclude that the improvement that we observe over the system VINCULUM, can for the most part be attributable to the weighted semantic relatedness measure. The fact that contrary to AIDA, PBoH and NCEL we do not construct the mention entity graph explicitly but rather rely on a set of independent contextual decisions—in other words, we use local optimization instead of global optimization of Eq. 3.4—also confirms this conclusion.

Statistical significance of the differences observed in Tab. 3.3 for methods present in GERBIL was assessed by means of a Student test for the Reuters 128 and RSS500 datasets. To this end, we built for each of these datasets 20 subsets of 20 randomly sampled documents. Micro-averaged error rates for each of the methods on each of the subsets are used in a paired t-test to compare methods, testing the equality of mean over two populations representing two CEL methods. For practical format reasons due to the GERBIL platform, statistical significance for the AIDA datasets could not be tested. Table 3.5 reports the test statistics T values for A *vs.* B combinations of methods: for

Approach	AIDA-A	AIDA-B	Reuters128	RSS-500
CNN [26]	-	85.5	-	-
End-to-End [41]	89.4	82.4	54.6	42.2
NCEL [13]	79.0	80.0	-	-
AGDISTIS [69]	57.5	57.8	68.9	54.2
Babelfy [54]	71.9	75.5	54.8	64.1
AIDA [34]	74.3	76.5	56.6	65.5
PBoH [28]	79.4	80.0	68.3	55.3
WSRM	90.6	87.7	79.9	79.3

Table 3.3 – Micro-averaged F1 score for different methods on the four datasets.

Approach	AIDA-A	AIDA-B
VINCULUM (Ref) [46]	69.1	66.4
VINCULUM (WLM) [46]	69.5	67.7
VINCULUM (both) [46]	69.4	67.5
WSRM	87.8	83.6

Table 3.4 – F1 score for VINCULUM (as reported in [46]) and WSRM on the AIDA datasets.

	Reuters128	RSS-500
WSRM vs AIDA	13.05	5.75
WSRM vs PBoH	4.27	14.33
PBoH vs AIDA	7.12	-6.78

Table 3.5 – t-values for the statistical significance test of A/B pairs using the micro-averaged F1 scores. Rejection region at a risk $\alpha = 5\,\%$ for the equality of mean between A and B is $T > 2.539$ for the one-tail t-test and $|T| > 2.093$ for the two-tail t-test.

an alternative hypothesis "A is better than B" (one-tail t-test), statistical significance is achieved if $T > 1.73$ with a risk $\alpha = 5\,\%$, with $T > 2.539$ with a risk $\alpha = 1\,\%$; for an alternative hypothesis "A and B are different" (two-tail t-test), statistical significance is achieved if $|T| > 2.093$ and $|T| > 2.861$ for $\alpha = 5\,\%$ and $\alpha = 1\,\%$ respectively. Values reported in Tab. 3.5 consistently demonstrate significant gains of the WSRM-based CEL method over AIDA and PBoH. PBoH and AIDA are also significantly different one from another, however with different conclusions depending on the dataset as explained above.

3.4 Conclusion

This chapter proposes to use the semantic interrelations between entities in a structured RDF KB for the collective entity linking. Approaches from the literature addressed this problem leveraging entity relatedness measures accounting for Wikipedia hyperlink graph which only expresses vague interrelations between entities. A few collective entity linking techniques advocated for structured RDF KB, limiting themselves to the existence of a relation between two entities in the KB at hand. We rather propose to use a structured RDF KB and weight relations between entities in the KB. Therefore, the new entity relatedness measure that we propose bears semantics since it uses relations between entities in BaseKB, a semantically structured KB, and attributes a weight to each couple of entities so that entities with a large total number of relations get large probability to be jointly mentioned in a document. This relatedness measure, combined with popularity and cosine similarity are the main ingredients we used to define a lightweight collective entity linking algorithm that was shown to compete with and outperform the state of the art in collective entity linking. We showed through experimental validation that it is feasible and beneficial to take into account the semantics of entities interrelationships for the collective entity linking.

The relatedness measure WSRM we proposed here is the cornerstone of our colective entity linking system. The good results we obtained with WSRM can be explained by the fact that this measure allows to incorporate all the direct relations between two entities in an RDF KB in one score that we use for the collective linking. Moreover, using WSRM enhances the use of highly connected entities in an RDF KB to perform the collective entity linking, which is inline with the collective hypothesis. This good behavior of WSRM is beneficial for any task that assumes the collective hypothesis.

These results beside showing the interest of using semantic RDF KBs, open new perspectives for taking into account the richness and expressiveness of structured KBs for entity linking, yet maintaining scalability. In particular, working on RDF KB, opens the door to fully exploit RDF KB semantics featuring semantic reasoning. While in the current chapter, we limited ourselves to direct relations between entities in RDF KBs to show the benefit of semantic KBs in the CEL process, one straightforward extension would be to consider also indirect relations between entities e.g. paths of length $m > 1$ between two entities in the KB. Nonetheless, this extension raises several challenges such as controlling the semantic drift when using paths with length greater than one,

or maintaining the scalability as the number of paths will quickly explode. We therefore propose in the next chapter to define the requirements for a good entity relatedness measure that will benefit the most from semantic RDF KBs and skirts the previous issues.

Chapter 4

Path-based Entity Relatedness Measures for Efficient Collective Entity Linking

In the previous chapter, we proposed a new semantic entity relatedness measure WSRM that accounts for the number of relations between two entities in RDF KB, and showed it to be beneficial for the collective entity linking, as it allowed to improve the performance of collective entity linking. This measure only considers the relations that *directly* connect two entities within the KB, i.e., considering property paths of length 1 in the RDF graph. Typically, WSRM only considers RDF triples directly linking e_i and e_j. For simplicity and tractability reasons, indirect connections between e_i and e_j, evidenced through property paths of length $m > 1$ in the RDF graph, are simply ignored, despite the valuable information they might provide. For instance, in Fig. 4.1 which depicts the properties in BaseKB related to a piece of text from the TAC-KBP2017 corpus (presented in Sec 3.3.1), the two entities 'New Jersey Legislature' and 'New Jersey' are not directly connected in the RDF graph but are indirectly connected through the entity 'New Jersey General Assembly' in a path of length two. While these two entities have a null relatedness score according to Ref or WSRM, they are clearly related, and a good entity relatedness measure must quantify to which extent they are related. Therefore, we argue in this chapter that indirect paths can be used to improve the quality of WSRM and the underlying collective entity linking system.

Incorporating indirect paths within an entity relatedness measure raises several challenges as explained in Chapter 3, particularly controlling the semantic drift —direct links should be prefered over indirect ones— and maintaining the scalability of the entity relatedness measure. We therefore propose in this chapter to study the requirements that a good entity relatedness measure must meet. The contributions we detail hereunder have been published in [22]. First, we discuss those requirements in Sec. 4.1. Then, we propose

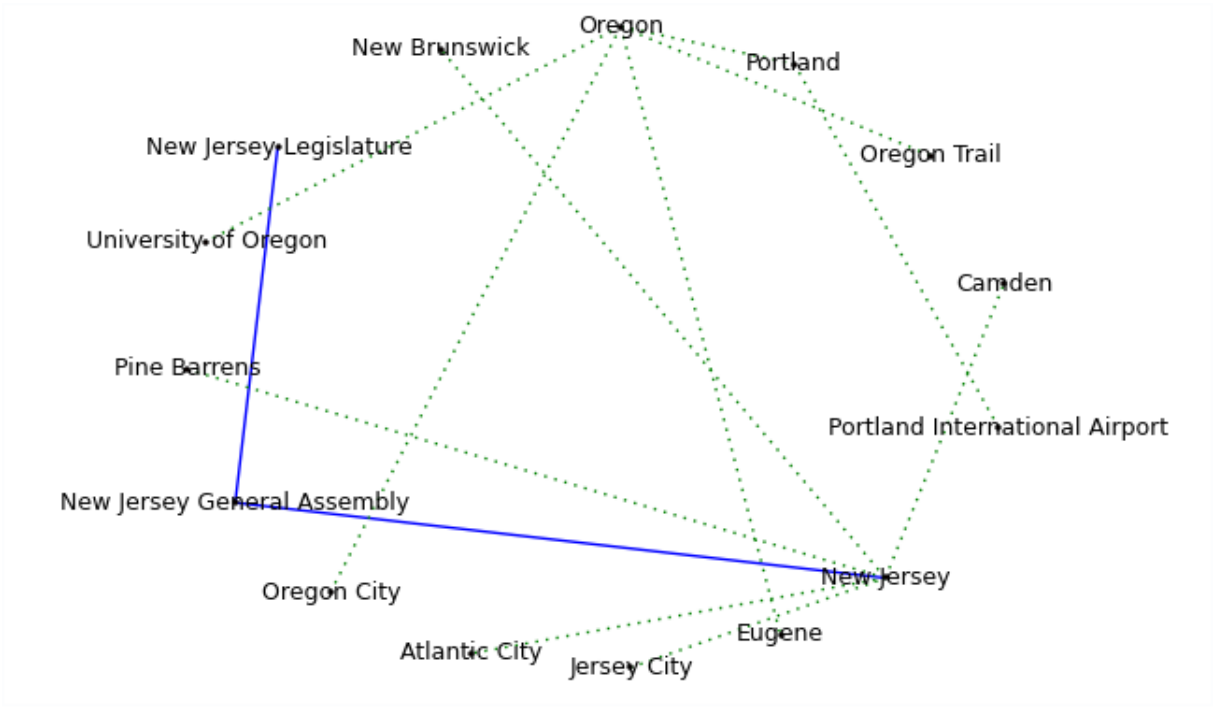


Figure 4.1 – Sample paths between candidate entities of entity mentions from TAC-KBP2017 dataset. Two entities are connected either with a solid line if their WSRM value is above 0.01, or with a dotted line if their non null WSRM value is below 0.01

in Sec. 4.2 a new path-based measure that at least partially meet those requirements. In Sec. 4.3, we describe a collective entity linking system with which the new path-based measure is experimentally compared to state-of-the-art competitors in Sec. 4.4. Finally, we conclude and discuss perspectives of this new measure in Sec. 4.5.

4.1 Requirements for a Well-founded Entity Relatedness Measure

In this chapter, we focus on incorporating (basic) reasoning mechanism by capitalizing as far as possible on indirect paths between the entities of an RDF KB. Hence, we study hereunder the requirements that a good entity relatedness measure should meet to improve the collective entity linking. Notably, we advocate that, in addition to showing significant performance improvement on standard benchmarks w.r.t. state-of-the-art competitors, a *well-founded* measure should meet the following three requirements to the extent possible: **(R1)** it must have a *clear semantics* so that linking decisions can be easily understood or

explained, in particular it must build on a knowledge base with formal semantics (e.g., an RDF or OWL one, as opposed to Wikipedia) and avoid tuning parameters or knobs that are hard to set by end-users, **(R2)** it must be calculated at a *reasonable computational cost* to be of practical interest and **(R3)** it must consider relatedness as a *transitive relation*, to capture entities that may be related within the KB either directly or indirectly (using composition of relations), i.e., through *paths*. The last requirement **(R3)** is crucial as it allows encoding implicit links between entities. For instance, if X *worksFor* Y and Y *isLocatedIn* Z then, the path from X to Z implicitly encodes X *worksIn* Z, which is an information not stored in the KB that can be captured by measures meeting **(R3)**.

To the best of our knowledge, no entity relatedness measure in the literature meets all these three requirements. Wikipedia based entity relatedness measures e.g., [1, 9, 16, 34, 28, 59, 43], consider Wikipedia's web page URIs as entities, web pages as textual entity descriptions, and hyperlinks between web pages as generic relations between entities. However, Wikipedia hyperlinks carry very loose semantics: it solely indicates that an entity somehow occurs in the description of another, be it central to this description or unimportant. Hence, Wikipedia-based entity relatedness measures do not meet **(R1)**, at least. A handful of measures relies on RDF KBs [1, 46, 36, 7, 61, 24]. Such KBs model both data (facts) and knowledge (ontological description of the application domain) using explicit and implicit triples; the latter can be derived through reasoning based on an RDF-specific consequence relation, a.k.a. entailment. In particular, within RDF KBs, the precise relation (a.k.a. *property*) r that directly relates an entity e_i to another entity e_j is encoded by the triple (e_i, r, e_j). The use of RDF KBs can therefore be seen as an important step towards devising well-founded entity relatedness measures. We recall below the few relatedness measures that use RDF KBs, and discuss to which extent they meet the three requirements of well-foundedness introduced above: **(R1)**, **(R2)** and **(R3)**.

The binary indicator Ref [1, 46] is defined between two entities e_i and e_j as:

$$\mathrm{Ref}(e_i, e_j) = \begin{cases} 1 & \exists r \; s.t. \; (e_i, r, e_j) \in \mathrm{KB}; \\ 0 & \text{otherwise.} \end{cases} \tag{4.1}$$

The above definition shows that Ref has a clear semantics **(R1)** as it is based on the existence of semantic relations in an RDF KB, and a low computational cost **(R2)** since it can be computed using edge lookups. We however remark that, though clear, its semantics is *very simple*: it does not take into account the various properties between e_i and e_j, nor those that e_i and e_j may have with other entities. Further, Ref does not allow

entities to be related through a property path within the RDF KB, hence does not meet **(R3)**: they can only be related through a *single* property, i.e., a single edge or triple.

The Weighted Semantic Relatedness Measure WSRM that we have proposed in Chapter 3, improves on Ref by not only accounting for the existence of some property between two entities using a Boolean value, but also by *weighting* how related they are in the [0,1] interval, assuming that the more properties between them, the stronger their relatedness.

The definition in Eq. 3.8 shows WSRM to have a clear and more fine-grained semantics than Ref **(R1)**, since it is based on all the relations between two entities and not only on the existence of one relation. Also, clearly, it can be computed at low computational cost **(R2)** based on edge lookups. However, like Ref, it does not allow entities to be related through property paths within the RDF KB, hence does not meet **(R3)**.

The path-based semantic relatedness measure [36] between two entities, denoted $rel^{(k)}_{Excl}$, is an aggregation of *path weights* for the top-k paths with highest weights between those entities; path weights are computed using the so-called *exclusivity* measure

$$\text{exclusivity}(x \xrightarrow{\tau} y) = \frac{1}{|x \xrightarrow{\tau} *| + |* \xrightarrow{\tau} y| - 1} \ , \tag{4.2}$$

where $|x \xrightarrow{\tau} *|$ is the number of outgoing τ relations for x, while $|* \xrightarrow{\tau} y|$ is the number of incoming τ relations for y; 1 is subtracted to avoid counting the relation $|x \xrightarrow{\tau} y|$ twice. Given a path $\mathrm{P} = x_1 \xrightarrow{\tau_1} x_2 \xrightarrow{\tau_2} ... \xrightarrow{\tau_{k-1}} x_k$ within the KB, its weight is

$$\text{weight}(\mathtt{P}) = \frac{1}{\sum_{i=1}^{k-1} 1/\text{exclusivity}(x_i \xrightarrow{\tau_i} x_{i+1})} \ . \tag{4.3}$$

Finally $rel^{(k)}_{Excl}$ is defined as the weighted sum of the top-k paths with highest weight between x and y

$$rel^{(k)}_{Excl}(x, y) = \sum_{\mathtt{P} \in P^k_{xy}} \alpha^{\text{length}(\mathtt{P})} \text{weight}(\mathtt{P}) \tag{4.4}$$

where P^k_{xy} denotes the top-k paths with highest weight between x and y, and $\alpha \in [0, 1]$ is a constant length decay factor introduced to give preference to shorter paths.

We remark that the above definition relies on paths between entities to measure their relatedness **(R3)**. However, we note that the semantics of $rel^{(k)}_{Excl}$ is controlled with param-

Measure	**(R1)**	**(R2)**	**(R3)**
Ref [1, 46]	×	×	
$rel^{(k)}_{Excl}$ [36]	∼		×
WSRM Chapter 3	×	×	
cosine [7, 61]	×	×	

Table 4.1 – Entity relatedness measures in the light of well-foundedness requirements: × indicates the requirement is met, while ∼ indicates it is only partially met.

eters whose "good" values are hard to guess, though $k = 5$ and $\alpha = 0.25$ are recommended default values based on empirical observations; thus $rel^{(k)}_{Excl}$ hardly meets **(R1)**. Further, the above definition requires to compute all the paths within the KB, which may not be computationally feasible since in large KBs, like the encyclopedic ones used for entity linking, the number of paths blows up as the considered path length increases; hence $rel^{(k)}_{Excl}$ does not meet **(R2)**.

Cosine similarity [7, 61] is used to measure the semantic relatedness between two entities in entity linking systems based on embeddings, e.g., [53, 59, 47, 13]: entities are mapped into coordinates of a multidimensional space, in which the closer two entities are, the more related they are (see Sec. 2.3). Several kernels exist for computing such embeddings, e.g., [7, 61, 57]. While the cosine similarity itself has a clear semantics **(R1)** and is not costly to compute **(R2)**, the machine learning-based construction of the entity embeddings cannot guarantee that cosine similar entities are indeed somehow related through some path in the KB, hence does not meet **(R3)**.

Tab. 4.2 recaps the above discussion and highlights that none of the entity relatedness measures used so far in the entity linking literature meets the three requirements of well-foundedness. Devising a measure that meets them all is a contribution of this chapter, which we present next.

4.2 The Path-based Weighted Semantic Relatedness Measure

Our approach to define a novel entity relatedness measure validating all the well-foundedness requirements extends a measure from the literature that only considers properties (direct relations) between entities, to a measure that considers paths between entities. In the sequel, we chose to rely on the measure WSRM that we have proposed in

Chapter 3, to capitalize (i) on properties **(R1)** and **(R2)** that WSRM verifies and (ii) on its state-of-the-art performance for collective entity linking see Sec. 3.3.

A straightforward extension of WSRM to take into account paths between entities would consist in counting the paths between the entities e_i, e_j and e_i, e', instead of the properties r and r' respectively in Eq. 3.8. However, the resulting measure would loose **(R2)** as it would require to compute all the paths between the entities in the KB. To circumvent this issue and retain **(R2)**, one may be tempted to only count paths up to some typically small length, as it is well-known (e.g., [36]) that the longer a path between two entities, the weaker the semantics of the relation it encodes. Still, in this case, though clear, the semantics of the resulting measure is poor as it does not account for the strength of the paths between entities.

Instead, in addition to bounding the length of the paths we consider, we do aggregate the WSRM values of the successive entity pairs found along a path between two entities, so that the resulting value reflects how related these entities are through this particular path. Further, since many paths (with same or different lengths) may relate two entities, we also aggregate the individual relatedness values of these paths into a final entity relatedness score. Hereafter, the aggregation operator for the WSRM values found along a path is denoted $\otimes$, while the one for path scores is denoted $\oplus$. Though typical candidate operators for $\otimes$ and $\oplus$ are either min and max, or product and sum, we chose fuzzy logic operators modeling the counterparts of the Boolean logical AND and OR operators in the [0,1] interval (recall that WSRM values are also within this interval). We now discuss three strategies to combine path relatedness values, yielding a family of entity relatedness measures.

The first strategy consists in aggregating all paths of length m separately, and aims at showing the contribution of paths with different lengths when considered separately. Formally, we define the weighted semantic relatedness measure for path of length m between entities e_i and e_j as

$$\text{ASRMP}^a_m(e_i, e_j) = \oplus_{p \in e_i \rightsquigarrow e_j, |p|=m} \otimes_{k=1}^{|p|} \text{WSRM}(p_k, p_{k+1}) \ , \tag{4.5}$$

where $e_i \rightsquigarrow e_j$ denotes the set of paths between e_i and e_j, here limited to paths of length m, and p_k is the k^{th} entity along path p (hence $p_1 = e_i$ and $p_{|p|+1} = e_j$). The inner $\otimes$ operator aggregates the WSRM scores along the edges of a given path; the outer $\oplus$ operator aggregates scores obtained for different paths of length m between the two

entities.

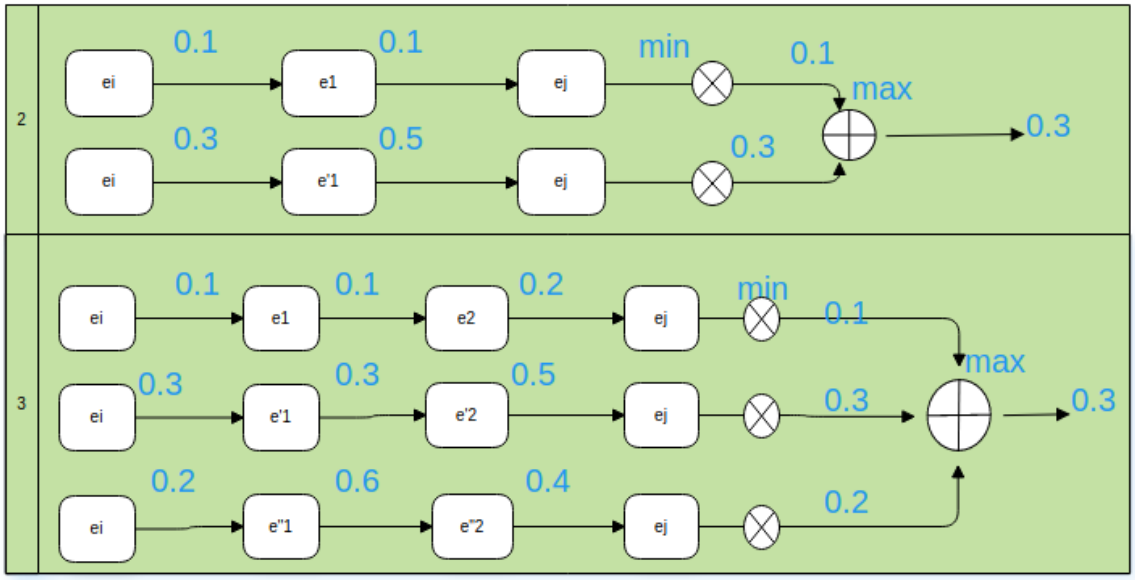


Figure 4.2 – Example of aggregation with ASRMP^a_m

The cost of the different aggregations is low, so $\text{ASRMP}^a_m(e_i, e_j)$ meets both **(R2)** and **(R3)**. It however only roughly meets **(R1)**, because the semantics is deteriorated by combining separately the paths of different lengths at a subsequent stage, e.g., in the entity linking process. We depict in Fig 4.2 an example of the aggregation with ASRMP^a_m, where min is used as $\otimes$ and max as $\oplus$. We can see that we have one score for each different path length that will be once again aggregated at classification time.

To avoid this two stage aggregation, a second strategy consists in aggregating all paths of length less or equal to m, as opposed to limiting ourselves to paths of a given length,

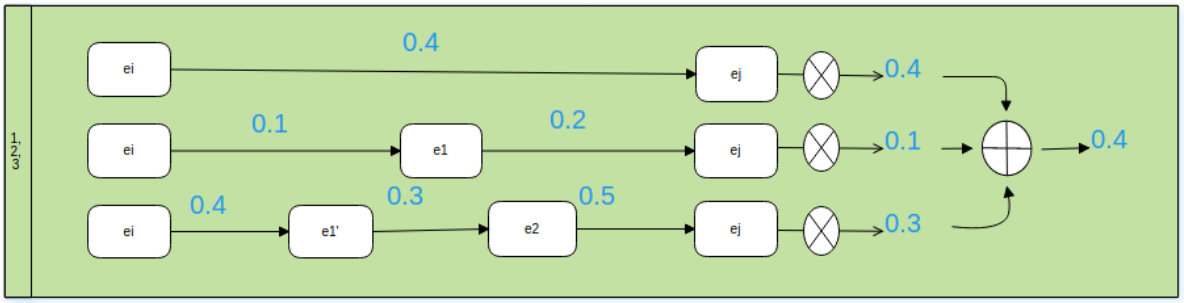


Figure 4.3 – Example of aggregation with ASRMP^b_m

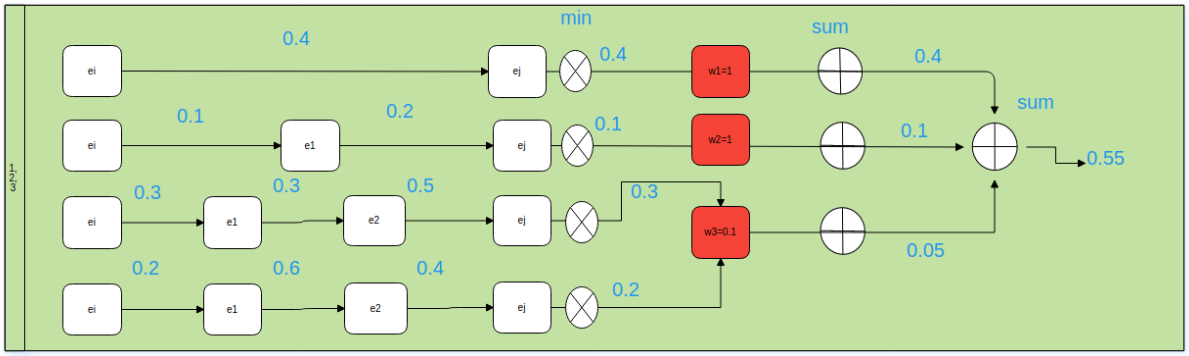


Figure 4.4 – Example of aggregation with ASRMP^c_m

extending Eq. 4.5 as

$$\text{ASRMP}^b_m(e_i, e_j) = \oplus_{p \in e_i \rightsquigarrow e_j, |p| \leq m} \otimes_{k=1}^{|p|} \text{WSRM}(p_k, p_{k+1}) \ . \tag{4.6}$$

In Fig 4.3, we depict an example of the aggregation with ASRMP^b_m, we can see that the aggregators *min*/*max* select the best path, which is the path of length one, and we explain to some extent the computed score.

The measure ASRMP^b_m, provides a first approach to combining paths of different lengths, however assuming equal weight for all of them. This assumption seems unrealistic: intuitively, direct relations are expected to account for strong relations, while indirect ones are weaker, where the longer the path, the weaker the relation. We thus introduce a weight depending on the path length according to

$$\text{ASRMP}^c_m(e_i, e_j) = \sum_{l=1}^{m} \sum_{p \in e_i \rightsquigarrow e_j, |p|=l} w_l \otimes_{k=1}^{|p|} \text{WSRM}(p_k, p_{k+1}) \ , \tag{4.7}$$

where w_l is a length-dependent weight roughly corresponding to the percentage of useful paths of length l and optimized by grid search. Thus, $\text{ASRMP}^b_m(e_i, e_j)$ meets the three requirements while $\text{ASRMP}^c_m(e_i, e_j)$ partially meets **(R1)**, because the semantics is once again deteriorated by the introduced weight. Fig 4.4 shows an example of the aggregation with ASRMP^c_m, all the paths of the same length are weighted and summed at the first *oplus* gate, and finally we just sum up the contribution of all the paths between the entities e_i and e_j, to evaluate their relationship.

Tab. 4.8 recaps the above discussion and shows that ASRMP^x_m meets partially **(R1)-(R3)**, since its semantic is slightly deteriorated by the classifier in ASRMP^a_m, and by the

Measure	**(R1)**	**(R2)**	**(R3)**
ASRMP^a_m	$\sim$	$\times$	$\times$
ASRMP^b_m	$\times$	$\times$	$\times$
ASRMP^c_m	$\sim$	$\times$	$\times$

Table 4.2 – ASRMP^x_m in the light of well-foundedness requirements: $\times$ indicates the requirement is met, while $\sim$ indicates it is only partially met.

weighting scheme in ASRMP^c_m.

Finally, all measures are made symmetrical according to

$$\psi^x_m(e_i, e_j) = \frac{1}{2}\left(\text{ASRMP}^x_m(e_i, e_j) + \text{ASRMP}^x_m(e_j, e_i)\right) \qquad x \in \{a, b, c\} \ . \tag{4.8}$$

The rationale for symmetrization is that in an RDF KB, if a triple (e_i, r, e_j) exists, the symmetric triple (e_j, r^-, e_i) may not exist at the same time, e.g., for r, r^- the symmetric properties 'hasWritten', 'writtenBy' respectively. This depends on the modeling choices adopted for the KB at design time.

Aggregating scores with fuzzy logic

The score aggregators used in the definition of ASRMP^x_m are crucial: they have to be chosen so as to preserve the semantics of the relations between entities without introducing noise, i.e., semantic drift. The longer a path between two entities, the smaller should be the relatedness value because the link between the entities may become meaningless. Typically, a product of WSRM values along a path will quickly decrease, resulting into useless scores; the average score can be noisy. For two given entities with a direct link and indirect links, the average can also result in scores for paths of length $m > 1$ larger than the score for the direct link, which we assume to be semantically incorrect. Hence we advocate for fuzzy logic operators which provide a wide range of aggregators, such as the equivalent of the AND/OR logic operators for real values in the $[0, 1]$ interval. The semantics of the fuzzy operators is also important because it allows to explain the linking decisions and ensures **(R1)**.

Fuzzy logic, especially triangular norm fuzzy logic (*t-norm*) which guarantees triangular inequality in probabilistic spaces, generalizes intersection in a lattice and conjunction in logic, offering many aggregation operators to define conjunction for values within $[0, 1]$. Each t-norm operator is associated with an s-norm (t-conorm) with respect to De Mor-

gan's law: $S(x, y) = 1 - T(1 - x, 1 - y)$. The t-norm is the standard for conjunction in fuzzy logic and thus the pair t-norm/s-norm acts as AND/OR operators on real values in $[0, 1]$. Thus using fuzzy logic to define our relatedness measure allows to ensure its transitivity —relations composition— by definition and avoids the introduction of arbitrary weighting parameters like in $rel^{(k)}_{Excl}$.

As $\text{WSRM}(e, e') \in [0, 1]$, any t-norm/s-norm pair can be used to aggregate values along one path of length m and across all paths between two entities. We experimented with several pairs of fuzzy operators: beside the classical min/max, we also consider the family of Hamacher t-norms (Hamacher product) defined for $\lambda \geq 0$ as

$$T_{\text{H},\lambda}(x, y) = \frac{xy}{\lambda + (1 - \lambda)(x + y - xy)} \ , \tag{4.9}$$

the family of Yager t-norms defined for $\lambda > 0$ as

$$T_{\text{Y},\lambda}(x, y) = \max \begin{cases} 0 \\ 1 - \sqrt[\lambda]{(1 - x)^\lambda + (1 - y)^\lambda} \end{cases} \tag{4.10}$$

and the Einstein sum

$$T_{\text{E}}(x, y) = \frac{xy}{1 + (1 - x)(1 - y)} \ . \tag{4.11}$$

The two families of t-norm used here are not exhaustive but generalize many t-norms: one can easily see that $T_{\text{H},2}(x, y) = T_{\text{E}}(x, y)$; $T_{\text{H},1}(x, y)$ is known as the product t-norm; $T_{\text{Y},1}(x, y)$ is the Łukasiewicz t-norm. We studied a large body of those operators and chose the one maximizing the accuracy of the collective linking system described hereunder.

4.3 Linking with entity relatedness measure

We study the interest of our new path-based entity relatedness measure in the context of entity linking. We recall that in a general collective entity linking pipeline, semantic relatedness measures between entities are used at the end of the process to globally select the best candidate entity for each mention. We follow the same strategy as in Chapter 3 and train a logistic regression, along with features describing the mapping between the mention and the entity, to predict whether an entity is a good match (1) for a mention or not (0).

We adopt the same entity linking pipeline as Chapter 3. We suppose a perfect entity

mention detection stage and we use Cross-Wiki [65] for the generation stage as described in Sec. 3.2.2. The final stage is the candidate selection stage, a.k.a. disambiguation, in which the best candidate is selected for each mention taking into account possible relations to candidates from other mentions. Hence, we account for the new path-based entity relatedness measure for the disambiguation stage.

In the remainder of this section, a document D is represented by its set of entity mentions, $D = (m_1, ..., m_n)$. For each mention m_i, $C(m_i) = (e_{i1}, ..., e_{ik})$ denotes the set of its candidate entities.

4.3.1 Knowledge Base

In the experiments described in this chapter, we focus on two RDF KBs, namely Yago [1] and BaseKB [2] see (Sec. 3.3.1), but however make use of Wikipedia for candidate generation for practical reasons, since the names of Wikipedia pages are meaningful unique identifiers unlike entities' labels in KB. Yago, derived from Wikipedia, WordNet and GeoNames, currently has more than 10 million subjects and contains more than 120 million facts. Within those two KBs, interrelationships between entities bear precise semantics as specified by their schema. Contrary to Yago, BaseKB is saturated, i.e., all facts are made explicit with property instances thus circumventing the need for reasoning mechanisms, in other words facts in BaseKB are explicit stored in the KB. As, for practical reasons, we take advantage of Wikipedia in the candidate generation step, a mapping between Wikipedia and Yago or BaseKB entities is maintained (since the candidate generation step is done on Wikipedia, while the linking is conducted on Yago and BaseKB). We also limit ourselves to entities appearing both in Wikipedia and in the RDF KB, resulting in approximately 2.5M entities in BaseKB and 3M entities in Yago.

Note that while BaseKB and Yago are used in this chapter, there are no conceptual limitations to those KBs, ASRMP$_m$ being able to account for any RDF KB schema.

4.3.2 Supervised Entity Selection

To select the best candidate entity e_{ij} for each entity mention m_i in a document in a collective manner, we adopted the same supervised approach as in Chapter 3, where a

1. https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago
2. http://basekb.com/

classifier is trained to predict whether a mention and a candidate entity are related (1) or not (0). We used a binary logistic regression, denoted logreg(), applied independently on each mention-candidate entity pair, selecting for a mention m_i the candidate entity with the highest response from the classifier, i.e., , $\hat{\jmath} = \arg\max_j \text{logreg}(m_i, e_{ij})$. We also experimented with different classifiers —see Sec. 4.4.3 for details— and the choice of a binary logistic regression is motivated by its simplicity and the fact that it turned out the best classification strategy, with respect to other classification strategies. In our collective setting, the classifier relies on features describing the similarity between the mention and the entity on the one hand, and, on the other hand, the relatedness of the candidate entity under consideration with the candidate entities from other mentions in the document. The latter accounts for the context and ensures the collective aspect of the linking.

For the similarity between the mention and the candidate entity, we considered the same two features used in Chapter 3 namely the cosine similarity between the vectors representations of the mention and of the entity name within Wikipedia, as obtained with word2vec [50](see Chapter 2), and the Wikipedia popularity as provided by Cross-Wiki.

For the relatedness of the candidate entity e_{ij} with candidate entities from other mentions, i.e., e_{pq} with $p \neq i$, we relied on an aggregation of the scores $\psi(e_{ij}, e_{pq})$ over the set of candidate entities $\cup_{p \neq i} C(m_p)$, thus providing a global measure of how e_{ij} relates to other entity propositions in D where $\psi()$ is an entity relatedness measure (e.g., $rel^{(k)}_{Excl}$, WSRM, $ASRMP^x_m$). This aggregation is different from the one used to design our relatedness measure. We used sum and maximum aggregation, which has proven efficient in previous work and has been previously introduced in Chapter 3. The sum aggregator is defined in Eq. 3.10, while the maximum aggregators are defined in Eq. 3.11. Recall that the two aggregators are complementary: the sum provides a global averaged view while the max values emphasize good matches. We showed in Chapter 3 that retaining the sum, max@1, max@2 and max@3 aggregators as global features for the logistic regression worked best for the relatedness measure $\psi^a_1()$. We therefore retained the same strategy for $\psi^a_2()$, and $\psi^a_3()$ resulting in a total of 12 global features—namely S_m, $M^{(1)}_m$, $M^{(2)}_m$ and $M^{(3)}_m$ for $m = 1, 2, 3$—to represent the relatedness of a candidate entity with other possible entities in D. Experiments with $\psi^x_m()$ with $x \in \{b, c\}$, i.e., where different path lengths are already aggregated within ASRMP^x_m, involve only 4 global features, i.e., sum, max@1, max@2 and max@3. Thus ASRMP^a_m leverages 12 global features while ASRMP^b_m and ASRMP^c_m only use 4.

4.4 Experiments

In the remainder of the chapter, we report on a set of experiments conducted to assess the benefit of our entity relatedness measure in a collective entity linking task. We are using different entity relatedness measures, within the same collective entity linking pipeline as described per Sec. 4.3. To assess the benefit of our entity relatedness measure, the gold-standard (where entity mentions are already annotated) of the TAC-KBP 2016-2017 datasets were used. Similarly to Chapter 3, the 2016 version was used to train the classifiers while the 2017 one served as test set. As the collective entity linking system is trained while only changing the entity relatedness measure, the linking accuracy can be used to evaluate the quality of the entity relatedness measure.

After providing implementation details in Sec. 4.4.1, selecting the best fuzzy aggregator in Sec. 4.4.2 and the best classification strategy in Sec. 4.4.3, we compare in Sec 4.4.4 the various flavors of ASRMP$_m$ seeking for the best one. The latter is compared to the entity relatedness measures used for entity linking in the literature in Sec. 4.4.5. Finally, we compare in Sec. 4.4.6 our collective entity linking system to a series of competing systems.

4.4.1 Implementation Details

Computing *all* the paths of length m between *every* pair of entities in the KB can be computationally expensive. For instance, in BaseKB, and after data cleansing, there are approximately 13M paths of length one and 46B paths of length two. We designed an efficient way of doing so, taking advantage of a relational database management system—which offers today much more tuning opportunities than RDF data management systems, e.g., various indices, clustered tables, etc.—to store edges and their semantic relatedness weights.

In PostgreSQL 11.2 [3], a table $\texttt{edges}(\texttt{e}_1, \texttt{e}_2, \texttt{v})$ is used to store the pairs of entities (e_1, e_2) directly connected through some property in the KB, along with the corresponding WSRM value v. This table is dictionary-encoded (entity names are replaced by integers) to save space and speed up value comparisons, indexed by (e_1, e_2) and (e_2, e_1) values to offer many options to the PostgreSQL optimizer. Limiting ourselves to path of length $m \leq 4$, the four tables $\texttt{path1}(\texttt{e}_1, \texttt{e}_2, \texttt{v}_1)$, $\texttt{path2}(\texttt{e}_1, \texttt{e}_2, \texttt{v}_1, \texttt{v}_2)$, $\texttt{path3}(\texttt{e}_1, \texttt{e}_2, \texttt{v}_1, \texttt{v}_2, \texttt{v}_3)$ and $\texttt{path4}(\texttt{e}_1, \texttt{e}_2, \texttt{v}_1, \texttt{v}_2, \texttt{v}_3, \texttt{v}_4)$ are efficiently created from the edge table using SQL queries, to

3. https://www.postgresql.org

represent paths of length 1, 2, 3 and 4 respectively. The entities e_1 and e_2 are restricted to the candidate entities for the entity mentions found in the TAC-KBP2016-2017 datasets: entities along the paths may however not be candidate entities. The values v_i are the WSRM values along the path.

In BaseKB, we obtained approximately 53K one-, 11M two- and 2B three-edges paths, from which we computed the various ASRMP$_m$, relatedness values. We were not able to compute paths of length four, as the number of paths exploded, due to the redundancy in BaseKB. The same process was applied to Yago and we obtained approximately 28K one-, 845K two-, 25M three- and 679M four-edges paths. Paths of length four could be computed due to the cleanliness and the higher structure of Yago, as it has deeper hierarchy when compared to BaseKB.

4.4.2 Comparing Fuzzy Logic Aggregators

One crucial issue for paths of length $m > 1$ lies in the aggregation of the semantic relatedness measure of each edge along the path and of the relatedness measure over multiple paths between two entities. ASRMP$_m$ reflects entity relatedness in the KB at hand: obviously, an aggregation of its values should reflect similar properties. Moreover, and in order to avoid a semantic drift, the resulting value of the aggregation for one path of length m must be smaller than that of a path of length $m-1$ since the latter bears stronger semantics. Finally, because there can be many paths between two entities, one needs also to aggregate the values of the different paths connecting two given entities.

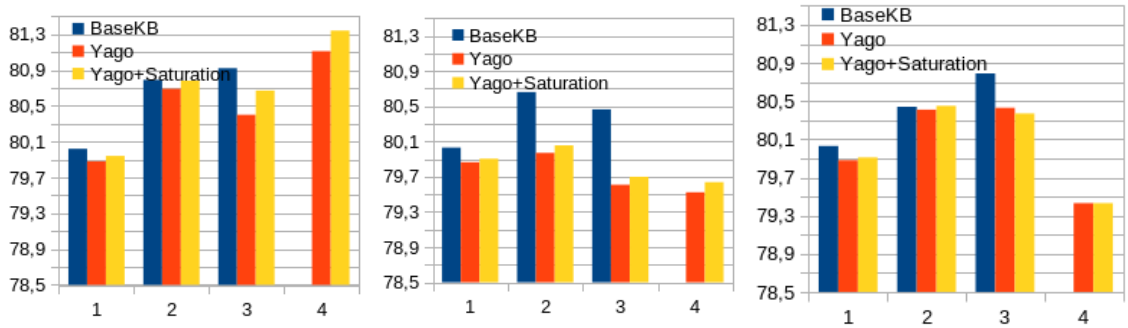
Experimental results show that $T_{\mathrm{H},0}(x, y)$ is the best aggregator with the collective linking setting in this chapter. We however experimentally observed only minor differences between the Hammacher and Yager t-norms and various values of λ. In the remainder, $T_{\mathrm{H},0}(x, y)$ with its associated s-norm is used for the aggregation of paths of length $m \in \{2, 3, 4\}$ between two entities.

4.4.3 Comparing Classifiers

We compared several classifiers within our collective entity linking system. In addition to popular classification techniques such as k-nearest neighbours (KNN), decision trees (DT), logistic regression (REG) or support vector machines (SVM), we also experimented with gradient boosting (GB). The latter was used in previous work on entity relatedness for entity linking [78, 79]. Results reported in Tab. 4.3 for ASRMP^a_m, $m \in \{1, 2, 3, 4\}$,

Approach	BaseKB					Yago					Yago+Saturation				
	KNN	DT	GB	SVM	REG	KNN	DT	GB	SVM	REG	KNN	DT	GB	SVM	REG
$ASRMP_1$	49.58	47.71	79.59	79.19	**80.03**	49.64	47.51	79.67	79.75	**79.88**	50.46	47.58	**80.05**	79.60	79.94
$ASRMP_2^a$	50.00	47.10	79.75	79.82	**80.79**	49.13	47.02	**80.93**	79.52	80.71	49.46	46.99	79.09	80.15	**80.78**
$ASRMP_3^a$	50.02	47.24	80.20	80.12	**80.60**	49.48	46.79	80.36	79.66	**80.40**	50.33	46.74	79.42	79.62	**80.67**
$ASRMP_4^a$	-	-	-	-	-	50.20	46.78	78.56	80.40	**80.98**	49.43	46.79	80.51	80.78	**81.34**

Table 4.3 – F1 scores for various classifiers within the entity linking system for TAC-KBP.



(a) classifier-based fusion (b) fuzzy logic fusion (c) explicit weights

Figure 4.5 – Linking F1 score for various aggregation strategies.

on the TAC-KBP dataset using either BaseKB or (saturated) Yago as KB, clearly show that the logistic regression classification strategy turns out to be the best option overall, in particular when considering paths of length 2 or more.

4.4.4 Comparing Aggregation Strategies

We also compared the aggregation strategies described in Sec. 4.2, reporting in Fig. 4.5 the F1 score as a function of m for the various strategies: distinct $ASRMP_m^a$ measures for each value of m (including length four for Yago) aggregated by the classifier; aggregation with fuzzy logic as defined by $ASRMP_m^b$; explicit weighting as in $ASRMP_m^c$ optimized by grid search. In most cases, better performance is achieved for $m \in \{2, 3\}$, diminishing for $m > 3$, which confirms that paths longer than 3 mostly bring noise because of the semantic drift. This is particularly visible in Fig. 4.5b. Classifier-based fusion, Fig. 4.5a, however seems to keep increasing for $m = 3$ on BaseKB, but the gain is only minimal between $m = 2$ and $m = 3$ and is counterbalanced by the computational cost (see Sec. 4.4.5), especially for BaseKB. Interestingly, for explicit weighting, the weights w_l can be seen as the strength of the paths with length l. We found that the optimal values of w_l decrease when l increases, i.e., $w_2 = 1$, $w_3 = 0.1$ and $w_4 = 0.1$ for Yago. These different aggregation

Table 4.4 – Linking F1 score on the TAC-KBP2017 dataset. Popularity and cosine similarity are the local mention-entity scores; the sum (S_m) and max ($M_m^{(k)}$) global features are defined in Eq. 3.10 and Eq. 3.11 resp.

Features	**BaseKB**	**Yago**	**Yago+Saturation**
local (no collective)	78.72	78.72	78.72
local+cosine similarity(rdf2vec)	78.58	78.58	78.58
local+cosine similarity(TransE)	79.39	79.39	79.39
local+Ref	79.70	79.81	79.82
$\text{local} + rel_{Excl}^{(5)}$	80.54	80.49	79.27
$\text{ASRMP}_1 = \text{local} + S_1 + M_1^{(k)}$	80.03	79.88	79.94
$\text{ASRMP}_1 + S_2$	80.02	80.02	80.12
$\text{ASRMP}_1 + M_2^{(k)}$	80.68	80.69	80.78
$\text{ASRMP}_2^a = \text{ASRMP}_1 + S_2 + M_2^{(k)}$	80.79	80.71	80.78
$\text{ASRMP}_2^a + S_3$	80.92	80.77	80.77
$\text{ASRMP}_2^a + M_3^{(k)}$	80.55	80.35	80.76
$\text{ASRMP}_3^a = \text{ASRMP}_2^a + S_3 + M_3^{(k)}$	80.60	80.40	80.67
$\text{local} + S_2 + M_2^{(k)}$	80.16	80.60	80.52
$\text{local} + S_3 + M_3^{(k)}$	80.42	79.46	79.27

studies show that fuzzy aggregator (Fig. 4.5b) and explicit weights (Fig. 4.5c) are more robust for combining paths of different lengths, while the classifier-based fusion (Fig. 4.5a) is more accurate though it introduces noise for paths of length > 2. For example, in both Fig. 4.5b and Fig. 4.5c paths of length four are always adding noise, when considered with Yago and Yago saturated. With respect to the entity linking task, ASRMP_m^a with classifier-based fusion appears the best strategy. In all generality and contrary to ASRMP_m^b, this strategy only loosely verifies **(R1)** as classifier-based fusion can be difficult to interpret. In this regard, logistic regression nevertheless offers interesting properties, with coefficients and intercepts that can be interpreted to some extent.

4.4.5 Entity Relatedness Results

We now concentrate on the study of (the different components of) ASRMP_m^a, $m > 1$, with classifier-based fusion, and how it compares with other relatedness measures, namely WSRM(see Chapter 3), cosine similarity [61, 7] and Ref [1, 46]. All measures are used within the same collective entity linking system as input features to the classifier, thus

providing fair comparison of the entity relatedness measures. Results are gathered in Tab. 4.4 for BaseKB, Yago and Yago saturated, reporting linking accuracy (F1 score). The different measures compared are:

— Local (no collective) performs linking using only the two local features depicting the adequacy of the mention and the entity—see Sec. 4.3.2—thus not considering entity relatedness
— Cosine similarity(kernel), the kernel being either rdf2vec [61] or TransE [7], measures entity relatedness as the cosine similarity between the entities embedded in a high-dimension space with the given kernel
— Ref [1, 46] considers the Ref entity relatedness measure as defined in Eq 4.1
— $Rel^{(5)}_{Excl}$ [36] uses entity relatedness as defined in Eq 4.4 with $k = 5$
— WSRM, which is equivalent to ASRMP_1, where only direct paths are used to measure entity relatedness
— ASRMP^a_m which embed basic reasoning mechanisms accounting for paths of length $m > 1$

Adding paths of length 2 allows a slight increase of the linking accuracy, where the best score for ASRMP^a_2 is obtained using both S_2 and $M^{(k)}_2$ for $k = 1, 2, 3$ (row ASRMP^a_2). Looking separately at the benefit of the aggregators S_2 and $M^{(k)}_2$ across a set of candidate entities, we see that considering only the maximum increases the accuracy of the ASRMP_1 system but, as it reflects the predominant topic, mentions that are far from that general topic can be incorrectly linked. Meanwhile, using S_2 can be slightly worse than ASRMP_1 only (e.g., on BaseKB, not on Yago) because this aggregator reflects choosing the mean topic which can be very vague. Combining both seems to be a compromise between the two extreme cases. On the other hand, ASRMP^a_2 is better than both WSRM and $Rel^{(5)}_{Excl}$ [36] showing the interest of using a well founded entity relatedness measure along with property paths.

Paths of length 3 can further be successfully combined with the features used for ASRMP^a_2 when S_3 is considered; while using $M^{(k)}_3$, either alone or with S_3, seems to introduce noise in the linking decision. This counter-intuitive result can be explained by the fact that introducing path of length three adds limited relevant semantics into the relatedness measure. As an outcome, considering the predominant entities only (max aggregators) tends to take strong linking decision and can be more drastic than adding vague links, mostly for entities that were not linked with the aggregation of ASRMP_1 and ASRMP^a_2.

	$TransE$	$rel^{(5)}_{Excl}$	Ref	ASRMP$_1$ = WSRM	ASRMP$^a_2$	ASRMP$^a_3$
BaseKB	15.29	1680	13.33	13.85	20,94	418,85
Yago	0.84	507	0.58	0.59	6.75	9.17
Yago+Saturation	0.79	403	0.57	0.69	6.14	8.60

Table 4.5 – Time in (min.) for different entity relatedness measures.

Approach	AIDA-A	AIDA-B	Reuters128	RSS-500
NCEL [13]	79.0	80.0	-	-
AIDA [34]	74.3	76.5	56.6	65.5
PBoH [28]	79.4	80.0	68.3	55.3
CEL-ASRMP$_1$ =CEL-WSRM	90.6	87.7	76.6	76.4
CEL-ASRMP$^a_2$	**93.8**	**91.0**	77.5	**76.6**
CEL-ASRMP$^a_3$	93.4	90.6	**78.5**	**76.6**
CEL-ASRMP$^a_4$	93.1	90.3	76.6	74.6

Table 4.6 – Micro-averaged F1 score for different collective entity linking systems on four standard datasets.

From the complexity point of view, relatedness measures are computed offline for a static KB (a given version of Yago or BaseKB). Meanwhile ASRMP^x_m can easily be computed for lower values of m making it tractable and more suitable for dynamic scenarii where entities are added to or removed from the KB, unlike rel^k_{Excl} where top-k paths with respect to Eq. 4.3 has to be computed, or cosine similarity where the kernel embedding has to be retrained. Tab. 4.5 shows the computation time for the different entity relatedness measures, including the offline part. For small values of m, which are required in practice, Ref, ASRMP^a_m, and $TransE$ have low computation cost, while rel^k_{Excl} has high computation cost due to the need to compute top-k best paths. Thus we can conclude that ASRMP^a_m meets **(R2)**, and more generally that ASRMP^x_m with $x \in \{a, b, c\}$ meets **(R2)**. They indeed have similar computation times: most of the time is spent in computing paths of length up to m, while aggregating path scores is very fast.

We also studied the impact of the saturation of the KB using Yago. As shown in Tab. 4.4 (columns 3 and 4) and in Fig. 4.5 (red and yellow bars), the gain is very limited in the case of TAC-KBP2017 dataset. In practice, this result saves the explicit computation of the implicit triples in the RDF KB.

4.4.6 Comparison of entity linking systems

We finally compared the collective entity linking system based on ASRMP^a_m with prominent state-of-the-art methods over standard benchmarks: NCEL [13], AIDA [34], PHoH [28] and CEL-WSRM. All follow the classical three stage architecture for collective entity linking. CEL-WSRM is the collective entity linking system proposed in Chapter 3, that is based on the WSRM entity relatedness measure (Eq. 3.8), equivalent to ASRMP$_1$. Results of the entity linking process, evaluated in terms of micro-averaged F1 classification scores, are reported in Tab. 4.6. These results were obtained with the Yago KB that allows considering paths of length up to 4. Similar results are obtained when the Yago KB is saturated. On all four datasets, the proposed method CEL-ASRMP^a_m, $m \in \{2, 3, 4\}$, does outperform the NCEL, AIDA and PBoH collective linking approaches by a large margin. The proposed method is better than CEL-WSRM on the four datasets, with small improvement on the RSS-500 dataset. Moreover, we observe the same conclusion as before: paths of length two improve the accuracy of the linking, while longer paths may add noise.

4.4.7 Beyond Figures...

Through an in-depth look into the results of our system for the TAC-KBP2017 dataset, we found that taking into consideration paths of length two leads up correcting some linking decision done with WSRM. For example, a mention 'Austin', incorrectly linked to 'Charlie Austin' (the English professional footballer) when paths of length one were solely considered, was correctly related to the capital of the U.S. state of Texas. Moreover, 53 mentions were newly linked to entities in BaseKB. Note that the way our classifier works does not allow us to know whether the added mentions correspond to paths of lengths one or two (or three), but solely that adding longer paths helps in the improvement of the global performance.

However we also found that using *max* and *sum* in the global features prevented the correct linking of 8 mentions that were correctly linked using WSRM. The same remark holds when combining both paths of length two and three with *max* solely in the global score: ASRMP$^a_3$ allows to correctly link 68 new mentions when compared to WSRM only; however it also increases the bad linking decisions to 11 instead of 8 before. But globally the number of linking decision errors that are corrected is higher than the one of those that are introduced. The upper bound limit is paths of length two for BaseKB and paths

of length three for both Yago and Yago saturated.

A deeper analysis of the remaining errors, that is, mentions for which considering longer paths still failed at establishing correct links, shows that most of them correspond to mentions that do not contain words of the name of their corresponding entity in the KB. e.g., *'CEO'*, *'city'* or *'police'*. The problem is especially severe within noisy contexts, that is, sentences or documents in which either several such mentions co-occur, or several co-occurring mentions refer to potential targets in the KB, making the collective disambiguation more challenging. If in *"Steve's a pretty unique CEO"*, *"CEO"* was correctly linked to *"Steve Jobs"*, in the more ambiguous *"In what seemed to be an even further distancing from the company's past, Chen was remarkably open to the idea of joining the competition"*, our system failed in linking *"company"* to the entity it refers to (*BlackBerry Ltd*), because that entity was mentioned together with several other names of companies somewhere before in the document.

4.5 Conclusion

In summary, we devised a new entity relatedness measure (ASRMP$_m$), that extends previous measures —and especially WSRM— by accounting for indirect relations between entities through the consideration of property paths. This measure is to the best of our knowledge the first to satisfy the three good properties that such measures should have: clear semantics, reasonable computational cost and transitivity. We showed the feasibility of incorporating indirect paths while maintaining the scalability and the semantics of ASRMP$_m$, leveraging fuzzy aggregators. We also experimentally showed the benefit of ASRMP$_m$ for the collective entity linking, where paths of length 2 and 3 bring improvement over the state of the art in collective entity linking, using either only direct connections between entities [24] or previous work on path-based relatedness measures [36]. In theory, the scalability of ASRMP$_m$ varies in inverse proportion with the length of the paths. We nevertheless, showed it to be still tractable for useful paths i.e., paths of length up to 3 as in practice longer paths add noise.

This contribution opens up new horizons towards fully exploiting the semantics of RDF knowledge bases for entity linking, based on relatedness measures. Our measure WSRM already accounted for semantic RDF KBs, and ASRMP$_m$ further introduced (basic) reasoning mechanisms that exploit the graph-structure of the KB leveraging robust aggregators for paths of arbitrary length. The main drawback of ASRMP$_m$ is selecting paths

regardless of their precise semantic. Hence, ASRMP$_m$ can be extended so that it consider only paths that are relevant for a given context, e.g., using ontological knowledge and reasoning. Moreover, the collective entity linking can also be restricted to a window-based context instead of the hole document. The idea here is to replace the global coherence of a document by an ensemble of small local contexts, as a mention used at the beginning of a document has small chance to be related to another one used at the end of this document.

Chapter 5

A correlation-based entity embedding approach for robust entity alignment

Up until this point, our main focus has been studying and designing entity relatedness measures for the collective entity linking, accounting for the precious semantics of RDF KB. We switch gears in this chapter and focus on a new area of knowledge extraction, namely *entity alignment*. The multiplicity of RDF KBs, not necessarily well-connected despite the huge efforts as part of the linked open data, however limits there practical use for the collective linking. Entity alignment, a.k.a. *entity matching*, which consists in discovering entities in two different KBs that refer to the same real-world concept, comes as a solution to this issue. Entity alignment is bound to facilitate a variety of tasks such as knowledge extraction and discovery, question-answering, semantic search, knowledge reasoning [8, 30], and entity linking introduced in the two previous chapters, with practical implications for companies wishing to complete their data warehouse with open data sources.

Entity alignment derives from ontology matching approaches [64]. The latter are based on matching symbolic features that describe entities [64], e.g., names, types or attributes, accounting for ontological knowledge. Symbolic approaches, suffer however from the semantic heterogeneity, introduced in particular by different languages and schemas. Hence, recently these methods have been superseded by learning-based approaches that seek to jointly embed entities from two KBs in a unique multidimensional space in which the actual entity matching is performed from the distance between entities of the two KBs [15, 81, 66, 67]. We further describe these methods in the Sec. 5.1.

Most of entity alignment techniques learn the joint embedding of the two underlying KBs donated KB_1 and KB_2, leveraging TransE criterion(see Sec. 2.3) for semantic embedding and a task-specific criterion, known as calibration, which ensures the coherence

of the joint embedding space. The latter guarantees that distance-based linking in the embedding space is meaningful and typically makes use of a *seed* set of known alignments between KB_1 and KB_2, a.k.a. prior knowledge, and a *ref* set of entities for which we are searching an alignment. Note that other entities might be present in either one of the KBs but are not considered in the alignment process. The *seed* alignment provides the initial links between the entities of the two KBs for a few examples, and allows to propagate the semantics of the KBs in the joint embedding space. In particular, the seed alignment is used to ensure that aligned entities lie close in the embedded space. Nonetheless, the seed alignment remains today the major bottleneck of state-of-the-art techniques: its size and quality have a substantial impact on the alignment accuracy, requiring human expertise in practice to create accurate seeds. Therefore, we propose in this chapter to define an entity alignment approach, that suppress the need for a high-quality seed alignment.

This chapter is organized as follows: Sec. 5.1 discusses the state of the art, introducing concepts and notations required for the description of our method in Sec. 5.2. Experimental results are grouped in Sec. 5.3 before concluding and discussing the perspectives of our approach in Sec. 5.4. [1]

5.1 Related Work and Notations

This section briefly introduces an overview of the prominent methods from the literature, before giving further insight on state-of-the-art iterative alignment methods. In the following, the *seed* and *ref* refer to a given set of aligned entities, i.e., a correspondance between an entity of KB_1 and one of KB_2. The *seed* is used for the training, while the *ref* is used for the evaluation.

5.1.1 Overview of existing work

Historically, automated entity alignment initially leveraged various symbolic features of KBs such as their properties and entities attributes. These approaches face the issue of heterogeneity between KBs, in particular different languages and schemas. To skirt the issue, a few approaches also make use of external lexicons, machine translation and Wikipedia links [64] to help match properties and attributes across KBs, yet remaining difficult to generalize and scale.

1. This work has been published in [21] in collaboration with Francois Torregrossa: francois.torregrossa@irisa.fr

The past few years have seen the fast emergence of embedding-based approaches based on representation learning and exhibiting better performance and generalization capabilities than symbolic approaches. All these methods combine knowledge graph embedding with an entity alignment objective function, leveraging two broad families of knowledge graph embedding techniques mainly based on TransE [7] and graph convolution networks (GCNs).

The first family is based on the KB embedding model TransE [7] (see Sec. 2.3), completed with an alignment objective function. We recall that TransE learns vector representations for entities and relations by enforcing the translation property, i.e., if the triple (h, r, t) hold in the KB, $v_h + v_r \approx v_t$ should also hold, where v_x denotes the embedding of a given element x. The resulting embedded space obviously bears the semantics of the KB, where translations in the embedded space correspond to relations in the KB. TransE is designed for the embedding of a single KB, which prevents from directly matching entities based on their distance. Therefore, most entity alignment techniques learn the joint embedding of the two underlying KBs donated KB_1 and KB_2, leveraging TransE criterion for embedding the two KBs in the same multidimensional space. MtransE [15] learns the embedding of each KB independently using TransE [7], and proposes different transformations to perform the alignment between the two embeddings. IPtransE [81] iteratively learns a joint embedding of KBs, based on PTransE [45]—a KB embedding algorithm similar to TransE—and integrates three modules (translation-based, linear transformation and parameter sharing) for jointly embedding different KBs. JAPE [66] further improves entity alignment by introducing attribute correlations in the process of KB embeddings. BootEA [67] interestingly proposes a constrained version of TransE, adding an explicit criterion to minimize the distance between aligned entities from the seed. BootEA also iteratively uses the links inferred between the KBs to progressively improve the alignment in a semi-supervised learning manner.

The second family leverages graph convolution networks (GCNs) instead of the TransE objective [76, 12, 77]. A straightforward, yet efficient, application of graph embedding is used in [76] to jointly embed the entities to align. MuGCN [12] additionally performs graph completion similar to KB saturation. Finally, RDGCN [77] builds a dual relation graph of the original KBs put together, a procedure similar to the notion of parameter swapping (see Sec. 5.1.2 for details on parameter swapping) and makes them interacting before jointly embedding entities through the dual graph. The construction of the latter renders the approach very sensitive to the quality of the seed, aligned entities in the seed

affecting the dual graph topology.

Globally, the main drawback of embedding-based techniques is the need for a high-quality seed, which is used to ensure the quality of the joint embedding space and, ultimately, maximize the accuracy of the final alignment on *ref*. As all methods directly and explicitly rely on the alignment provided in the seed, they are sensitive to the size and quality of the seed. On the other hand, the *ref* alignment is a set of entities from the two KBs for which we search a pairing. In academic work, the alignment on the *ref* set is known and used for evaluation purposes. Learning-based alignment techniques indirectly make assumptions on the *ref* alignments. For instance, the iterative approach [67] assumes uniform distribution of entity distances in *ref*, meaning that an entity from KB_1 in the *ref* alignment has the same probability to be aligned with any entity from KB_2, that is also in the the *ref* alignment. Similarly, [77] performs negative sampling only for entities in the *ref*. Clearly, the hypothesis claiming that distance between pairs of entities not in the *seed* should be uniformly scattered is wrong for truly aligned entities, and thus potentially harmful.

5.1.2 Background notations and technical details

We now provide further technical details on state-of-the-art embedding-based alignment, which we will make use of in the experimental section, introducing notation and highlighting the limits that we address.

Entity embedding objective function

Formally, the KB embedding objective function, which takes care of the semantics of the KBs by preserving semantic relations between entities, is defined similarly to TransE [7] (Eq. 2.3) as

$$O_e = \sum_{\tau \in T^+} [f(\tau) - \lambda_1]_+ + \mu_1 \sum_{\tau' \in T^-} [\lambda_2 - f(\tau')]_+ \ , \tag{5.1}$$

where $f()$ is a triple scoring function, here $f((h,r,t)) = ||v_h + v_r - v_t||_2^2$, and $[x]_+ = max(x, 0)$. T^+ and T^- denote the sets of positive and negative triples respectively. In plain words, O_e enforces the TransE translation property in the embedded space, minimizing the cost on positive triples—those bearing an actual relation—and maximizing on negative triples. In practice, *parameter swapping* is used as a standard practice to enforce similar

properties between the two KBs, augmenting the set of positive triples T^+ by injecting triples of KG_1 into KG_2 and vice-versa based on the seed alignment [67, 66, 81].

Alignment objective function

For alignment purposes, the objective function O_e is combined with an objective function O_a that measures the discrepancy between the vectors of the aligned entities, i.e., between the embedding of a vector of KG_1 and the embedding of the corresponding vector in KG_2. The actual form of the alignment objective function depends on the method [15, 45, 81, 66, 67]. Yet, all make direct use of the seed alignment and indirect assumptions on the alignment of the *ref* entities. For entities in the seed, the distance between the respective embeddings of the entities is minimized, typically considering a cosine distance. For entities in the *ref*, the objective function typically seeks to uniformly maximize the distance between any pair of entities, leveraging negative sampling in [77] or a likelihood matrix in [67]. This general scheme emphasizes two major limitations that we address in this chapter: (a) the distance between seed entity pairs is minimized batch-wise or individually, which prevents from modelling global alignment between embeddings and makes the approach sensitive to errors in the seed; (b) the hypotheses on other entities, which mix entities that appear in the two KBs and should thus be aligned with entities that have no counterpart in the other KB, are too strong, considering the two types of entities on equal foot.

Let us consider two examples of popular alignment objective functions leveraging BootEA [67] from iterative approaches and RDGCN [77] from GCN-based approaches, to further illustrate the previous statements. In the case of BootEA, O_a is formalized as follows:

$$O_a = -\sum_{x \in X} \sum_{y \in Y} \phi(x, y) \log \pi(v_x, v_y) \ , \tag{5.2}$$

where X is the set of entities from KG_1 to be aligned with Y from KG_2 and $\pi(v_x, v_y)$ is a measure of the similarity between the embeddings of x and y. The assignment function $\phi(x, y)$ is 1 if (x, y) are known to be aligned (as part of the seed or as given by semi-supervised learning —see below) and $\frac{1}{|Y'|}$ otherwise, $Y' \subset Y$ being the set of unaligned entities in Y. It basically indicates the probability that x corresponds to y or, similarly, that y is the label for x, considering non-aligned entities —entities in Y'— on equal foot which maybe is harmful as pointed out previously.

In the case of the GCN-based approach RDGCN, O_a is defined as follows:

$$O_a = \sum_{(p,q)\in S+} \sum_{(p',q')\in S-} [\lambda + ||v_p - v_q|| - ||v_{p'} - v_{q'}||]_+ \ , \tag{5.3}$$

where v_x is the embedding of the entity x obtained with the GCN, $\lambda > 0$ is a margin hyperparameter, and $[x]_+ = max(x, 0)$. S denote the seed entities and S^- denote the set of negative instances. S^- is obtained by replacing in (p,q) p or q with its nearest-neighbor entity in the embedding space, a.k.a. negative sampling. In plain words, O_a minimize the cost on positive pairs —those aligned— and maximizing on negative pairs. The negative sampling hypothesis is a standard practice at training time as in Eq. 5.3, but can be harmful at test time as pointed out previously(see Sec. 5.1).

Iterative alignment

Iterative alignment, a.k.a. *alignment bootstrapping*, casts the problem of entity alignment in a semi-supervised learning approach that alternates embedding learning and alignment inference from the embedding. For each training iteration, given a current embedding obtained from the combination of the two objective functions discussed above, the idea is to choose the most confident matching pairs to predict an alignment of entities in *ref*. A one-to-one mapping constraint is added in [67] and used in this work to improve the alignment prediction. This predicted alignment is then used in addition to the seed alignment in the reestimation of the embedding, impacting parameter swapping and the optimization of the combined objective function.

5.2 AlignD

To address the limits of the alignment objective functions that we presented in the previous section, we propose a novel alignment criterion that globally considers seed entities rather than iterating through the pairs of aligned entities, and that do not make assumptions on the *ref* entity pairing. The general idea of our approach relies on Pearson's correlation coefficient between the embedding of aligned entities along the same dimensions of the multidimensional space, with the idea of globally maximizing the correlation between corresponding dimensions as an indirect way to move the seed's aligned entities closer in the embedded space. This idea was inspired by [68] which tries to measure the global quality of word embedding techniques through dimension correlation measures.

Our aim is to compel dimension alignment of both embeddings such that aligned entity pairs are easily identifiable through cosine similarity by enforcing embedding dimensions to represent identical latent information and disentangle different dimensions.

The use of the global and indirect criterion offers increased robustness to seed alignment and removes assumptions on *ref* entities. This enables the investigation of automatically generated seed, hence containing errors, where most methods so far investigate the case of perfect seeds. We show how the proposed alignment objective function can be used within a robust iterative entity alignment system called AlignD and take advantage of its robustness to design a fully automatic entity alignment approach. The latter relies on an automatically generated seed, thus getting rid of the costly human intervention for the generation of an accurate seed.

5.2.1 Measuring Embedding Alignment

At the core of our approach is a measurement of how well the embeddings —E_1 and E_2 both with dimension D— of KG_1 and KG_2 match by looking at the correlation between dimensions of the embeddings over the matching entities in the seed. To this end, we define the correlation between two dimensions in the embedded space as Pearson's correlation between the coordinate of entities on the first dimension and the coordinate of the aligned entities on the second dimension. In other words, we quantify the dependency between the value of an entity embedding on the i-th dimension and the value of the corresponding (aligned) entity on the j-th dimension. The seminal idea, deriving from the fact that we use cosine similarity between embeddings to perform entity alignment, is that corresponding dimensions in E_1 and E_2 should match and carry the same information, which we designate as dimension alignment.

As we are interested in dimension alignment between the embeddings of the two different KBs, we rely on the correlation matrix where rows are indexed by the dimensions of the first embedding and columns by the dimensions of the second one. Entities are identified according to a seed set S of size N, such that the correlation coefficient at coordinates i, j models the interaction between the values of N entities of KG_1 on dimension i and the values of the corresponding N entities of KG_2 on dimension j. The idea is that dimensions i from KG_1 and j from KG_2 are likely to be aligned if the values of entities on those dimensions are significantly proportional. Formally, we write the correlation matrix A_D as

$$A_D = (r(E_1(S)_i, E_2(S)_j))_{(i,j)\in[\![1,D]\!]^2} \ ,$$

where $r(E_1(S)_i, E_2(S)_j)$ denotes the Pearson's correlation coefficient between dimension i of E_1 and j of E_2 on the entities from the seed set S.

Following this line of thought, a global indicator of dimension alignment quality is derived from A_D by observing two simple facts: on the one hand, E_1 and E_2 are aligned if dimension i in the embedding E_1 corresponds to dimension i in the embedding E_2, i.e., they are positively correlated; on the other hand, correlations on identical dimensions should be preponderant with respect to correlations between distinct dimensions. In other words, non diagonal terms in A_D should be small with respect to diagonal terms and hence the distance between A_D and I, the identity matrix, provides an approximation of the dimension alignment between the embeddings of the two KBs. In this work, we define the following criterion based on the L2-norm as a measure of dimension alignment discrepancy

$$O_d = \|A_D - I\|_2 \ . \tag{5.4}$$

High values of O_d means large deviation from the situation above and hence poor correlation between corresponding dimensions in the two embeddings, dimension alignment being obtained by minimizing O_d at learning.

O_d looks at vector coordinates for every pair in S simultaneously on each individual dimension through Pearson's correlation, thus forcing the representations to be globally consistent. The aim of O_d is thus twofold: (a) it ensures that each dimension of the embeddings E_1 and E_2 is consistent and avoids two distinct dimensions to be proportional, encouraging the encoding of distinct information in distinct dimensions; (b) it reduces the sensitivity to noisy pairs by processing all pairs in S at once, erroneous examples (if in minority) thus tend to be ignored.

Pearson's correlation is known not to be very robust to outliers, thus, erroneous aligned pairs might have an heavy impact on A_D. However, this is not a major issue in our case for the following reasons: (1) we want to minimize O_d, therefore, the value of the correlation is not important, only the maximal correlation achievable with erroneous pairs matters; (2) outlier terms are present both at numerator and denominator in Pearson's correlation coefficient, and their respective impact partially cancel each other; (3) if erroneous pairs are in minority, the effect on Pearson's correlation may be small. The robustness of AlignD with regards to erroneous pairs will be investigated in the experimental section.

5.2.2 AlignD Objective Function and Algorithm

Based on the measure of dimension alignment defined by Eq. 5.4, we propose a novel algorithm based on an iterative alignment philosophy of [67]. In the AlignD entity alignment system, the joint embeddings of the two KBs is obtained by minimizing the combined objective function

$$O = O_e + \alpha O_d \ , \tag{5.5}$$

where O_e is the TransE entity embedding objective function as defined in Eq. 5.1, O_d is the correlation-based alignment objective function as defined in Eq. 5.4 and α is a balance hyper-parameter. The combination of this new objective function O from Eq. 5.5 with parameter swapping and iterative alignment, as described in Sec. 5.1.2, leads to our alignment procedure AlignD. Both bootstrapping and the final alignment are computed leveraging the cosine similarity between embeddings. It is worth noting that the objective function O makes use of all entities in the embedding objective O_e yet not making specific assumptions for entities not present in the seed as O_d is obtained only from the seed alignments. We built the AlignD algorithm as detailed in Algorithm 1, that integrates KB embeddings, bootstrapping, parameter swapping and our dimension alignment criterion.

It is known that the size of the seed substantially influences the alignment accuracy, in particular, increasing the seed's size is likely to improve alignment of *ref* entities. This is particularly true for AlignD which does not put constraint on entities not in the seed, contrary to other methods. Hence, increasing the seed size has no side effects on the remaining entities and can only be beneficial. This suggests that the seed can be extended with automatic alignment methods, yielding a larger seed however with a small amount of incorrectly aligned entities. Because of the global vision of all entities in the seed (no batching) through O_d and of the absence of hypotheses on entities not in the seed, AlignD is designed to make the most of a large but not fully accurate seed, as long as the fraction of incorrect alignments in the seed remains reasonable, since the alignment is performed globally. On the contrary, other embedding-based alignment methods, such as BootEA or RDGCN, would be greatly affected by an extended or noisy seed since they enforce aligned entity vectors to perfectly match and assume uniformity of similarities for entities not in the source seed.

Algorithm 1: AlignD

Data: Triples from KG_1, noted T_1 and KG_2, noted T_2; a seed set S; a set of entities to be aligned A; Training parameters P

Result: The aligned embeddings E_1 and E_2 for KG_1 and KG_2 and a set of aligned entities R containing entities from S and A.

R := S;
E_1 := random_init_kg_embedding(extract_entities(T_1), extract_relations(T_1));
E_2 := random_init_kg_embedding(extract_entities(T_2), extract_relations(T_2));
T_1' := T_1;
T_2' := T_2;
for i :=1 to number_of_epochs*(P)* **do**
    # Perform parameter swapping to enrich triples with the aligned entities in R
    T_1' := extend_triples(T_1', R);
    T_2' := extend_triples(T_2', R);

    # Apply O_e objective function from Eq.5.1
    E_1 := embed_triples(E_1, T_1', P);
    E_2 := embed_triples(E_2, T_2', P);

    # Apply O_d objective function from Eq. 5.4
    E_1, E_2 := align_dimensions(E_1, E_2, S, P);

    # Bootstrap aligned entities set R with entities from A
    R := bootstrap_alignment(E_1, E_2, R, A, P);

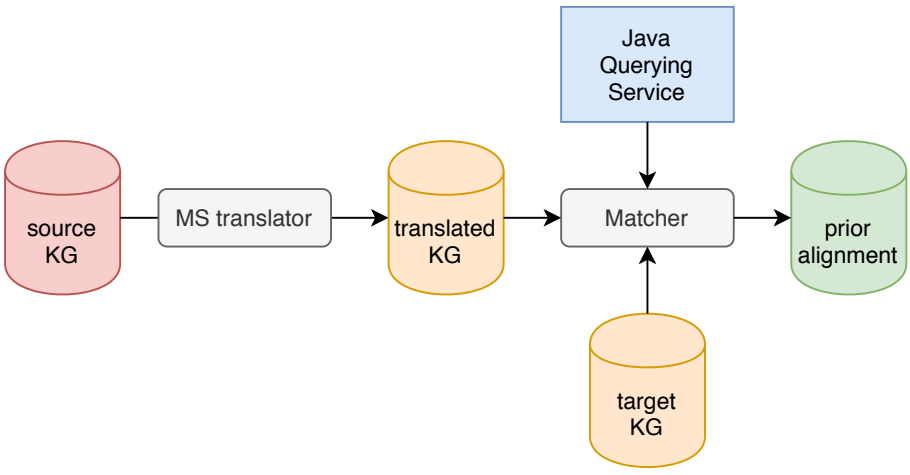


Figure 5.1 – Pipeline used to automatically compose a seed.

5.2.3 Automatic Seed Replacement

Matching KBs using embedding methods fail to provide an accurate alignment without a seed. Pushing the idea of automatically extending the seed alignments to an extreme case, we propose to replace the expert-based generation of a seed with a fully-automatic approach where the seed is generated automatically. In particular, we study a set of symbolic alignment approaches, which do not need manually-generated seeds, and use their predictions as a starting point in AlignD. Those symbolic matching approaches rely on basic string comparison between entities' names. The whole idea is to select the pairs of entities from the KBs that have strong similar labels according to some string metric. The process to automatically align entities from their labels, and thus predict a seed, is shown in Fig. 5.1, where we fix KG_1 to be the source KB and KG_2 the target one. Since the KBs are not necessarily in the same language, a translator module is used on the source KB, namely Microsoft Translator in our experiments. For every entity from the source KB, a list of entities from the target KB with a similar name is retrieved using the RDF query language SPARQL [2], and the one with the highest string similarity is selected, provided the similarity is higher than a threshold to keep only relevant matches. The threshold is set experimentally, when needed (some method do not require a threshold such as JaroWinkler), and depends on the string comparison technique used. Beside the standard string matching, we tried fuzzy matching to increase the recall of predicted aligned entities. We also studied four other string metrics, namely the Sørensen–Dice coefficient (DSC), similar to a Jaccard index, the Levenshtein distance, the JaroWinkler distance that gives more favorable ratings to strings matching from the beginning, and the cosine similarity using a pre-trained word embedding. This methodology, combining embedding-based entity alignment and symbolic approaches, allows KBs alignment without prior knowledge.

5.3 Experiments

Experimental validation is conducted on standard benchmarks, comparing AlignD with the state of the art and demonstrating its robustness with respect to the seed.

Table 5.1 – Statistics on datasets of DBP15k and DWY100K.

Datasets		#Entities	#Relations	#Triples
DBP-WD	DBpedia	100,000	330	381,166
	Wikidata	100,000	220	789,815
DBP-YG	DBpedia	100,000	302	451,646
	Yago3	100,000	31	118,376
ZH-EN	Chinese	66,469	2,830	379,684
	English	98,125	2,317	567,755
JA-EN	Japanese	65,744	2,043	354,619
	English	95,680	2,096	497,230
FR-EN	French	66,858	1,379	528,665
	English	105,889	2,209	576,543

5.3.1 Experimental Setup

The following standard datasets have been used in our experiments:

— DBP15K [66] contains three cross-lingual datasets built from the multilingual versions of DBpedia: Chinese to English (ZH-EN), Japanese to English (JA-EN) and French to English (FR-EN). Each dataset contains 15,000 aligned entities.

— DWY100K [67] contains two large-scale datasets extracted from DBpedia, and either Wikidata or YAGO3, denoted by DBP-WD and DBP-YG. Each dataset has 100,000 aligned entities.

Basic statistics on the datasets are reported in Tab. 5.1.

Following previous work on entity alignment, e.g., [66, 67], KB embeddings are trained with a seed containing 30 % of the alignments, the remaining ones being used for test purposes (ref). For evaluation, we classically report the mean reciprocal rank (MRR) and Hits@k after ranking all entities of KG_2 for each given entity of KG_1 according to the cosine similarity: Hits@1 strongly correlates with the quality of the alignment, while Hits@10 and MRR mostly indicates the quality of the embedded space for the alignment process. Higher Hits@k and MRR indicate better performance.

Three variants of AlignD have been implemented:

1. AlignD or AlignD(*glove*,300), where a perfect seed is used for the training, allowing to compare AlignD with state-of-the art approaches. See details on AlignD(*glove*,

2. https://www.w3.org/TR/sparql11-query/

Table 5.2 – Hits@1, Hits@10 and MRR with ground-truth seed on DBP15K datasets. The top half results are taken from the literature while we produced bottom ones by running new experiments.

Approaches	ZH-EN			JA-EN			FR-EN		
	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR
MTransE [15]	30.83	61.41	0.364	27.86	57.45	0.349	24.41	55.55	0.335
IPTransE [81]	40.59	73.47	0.516	36.69	69.26	0.474	33.30	68.54	0.451
JAPE [66]	41.18	74.46	0.490	36.25	68.50	0.476	32.39	66.68	0.430
MuGCN [12]	49.56	87.03	0.621	50.10	85.70	0.621	49.50	87.00	0.621
BootEA	61.89	84.01	0.695	57.43	82.93	0.661	58.31	84.83	0.676
AlignD	62.68	84.70	0.701	58.88	83.06	0.671	60.77	85.30	0.691
RDGCN	70.75	84.55	–	76.74	89.54	–	88.64	95.72	-
AlignD($glove, 300$)	82.30	93.93	0.864	84.90	94.24	0.882	90.85	97.26	0.913
BootEA($glove, 300$)	83.11	93.84	0.869	84.60	93.93	0.879	92.38	97.56	0.942

300) in Sec. 5.3.2.

2. AlignD[X,seed] where AlignD is trained with a seed obtained from the extension or the corruption of the ground-truth seed with an algorithm X, with the goal of comparing the robustness of AlignD and methods from state-of-the-art on the same setup.

3. AlignD[X] where AlignD is trained with a seed automatically generated by an algorithm X, to study the robustness of AlignD w.r.t. the presence of erroneous pairs in the seed, as well as its performance without prior knowledge.

For all experiments, the the best hyper-parameters [67] were chosen for KB embeddings with AlignD: $\lambda_1 = 0.01$, $\lambda_2 = 2$ and $\mu_1 = 0.2$. The parameter α depends on the seed and is set to the size of the seed at hand. The learning rate was set to 0.01, the training to 500 epochs, with semi-supervised alignment bootstrapping every 10 epochs. Those hyper-parameters are chosen according to previous experiment settings reported in the literature, notably BootEA [67]. Finally the embedding dimension was set to 75, except for AlignD($glove, 300$) and AlignD[RDGCN,seed] where the embedding dimension is 300. This is due to the pre-trained embeddings used in RDGCN [77], for which very few information is provided.

Table 5.3 – Hits@1, Hits@10 and MRR with ground-truth seed on DWY100K datasets.

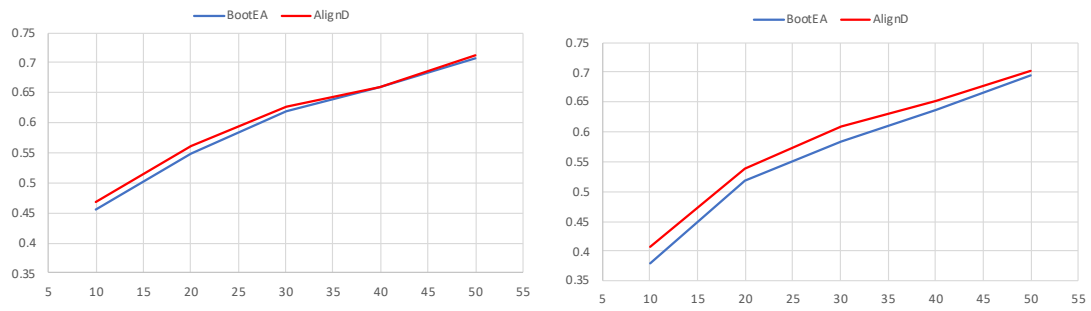
Approaches	DBP-WD			DBP-YG		
	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR
MTransE [15]	28.12	51.95	0.363	25.15	49.29	0.334
IPTransE [81]	34.85	63.84	0.447	29.74	55.76	0.386
JAPE [66]	31.84	58.88	0.411	23.57	48.41	0.320
MuGCN [12]	61.60	89.70	0.714	74.10	93.70	0.810
BootEA	70.52	86.00	0.75	61.74	77.33	0.671
AlignD	70.32	86.06	0.758	63.28	78.75	0.686

5.3.2 Comparison to State-of-the-Art Approaches using a Ground-Truth Seed

We first compare our approach to a series of entity alignment systems which reported state-of-the-art results in recent years on the DBP15K and DWY100K datasets, namely:

— MTransE [15] which learns a transformation between two fixed embeddings;
— IPtransE [81] which iteratively learns joint embeddings of KBs using PTransE [45];
— JAPE [66] which learns the embedding of KBs jointly while preserving entities attributes;
— GCN-based approaches MuGCN and RDGCN [12, 77], where the former uses graph completion to improve the matching while the latter builds a dual of KG_1 and KG_2 unified. RDGCN is our GCN baseline;
— BootEA [67], a semi-supervised technique which iteratively labels KG_1 entities with KG_2 entities, and which we consider as our iterative baseline.

All results are gathered in Tab. 5.2 and 5.3, using for all methods the same error-free seed corresponding to 30 % of the existing alignments. Results in rows 1 to 6, with embedding dimension 75, clearly show AlignD to be comparable or slightly better than the BootEA baseline on all datasets. RDGCN is not directly comparable to the other approaches in Tab. 5.2, as it uses pre-trained word embeddings of dimension 300, used to initialize embeddings of entities based on their name. To allow fair comparison, we thus trained AlignD using the same initialization as RDGCN, denoted AlignD(*glove*,300). In comparable conditions, AlignD performs significantly better than RDGCN. Experiments for RDGCN are however only provided on DBP15k as there are neither results in the litterature, nor pre-trained GloVe embeddings provided for the DWY100K datasets. We



(a) ZH-EN (b) FR-EN

Figure 5.2 – Hits@1 on entity alignment w.r.t. the seed size, expressed as the proportion of alignments used as seed.

also provide the results of BootEA with Glove-based initialization (BootEA(*glove*,300). BootEA(*glove*,300) and AlignD(*glove*,300) got very close results. Yet, we recall that pre-trained word embedding initialization is, in practice, impossible to reproduce for real-world KB, since it is difficult to find plain text covering their specific domain. For such reasons, we consider, in the following, results without Glove initialization to be more significant and more relevant. We still report the results using Glove to indicate the ideal performance.

Due to the computational cost of experimenting on DWY100, we limit experiments from now on to the dataset DBP15K, since we did not observe a significant differences in behavior in this series of experiments.

5.3.3 Impact of the seed size on performance

As known from the literature on entity alignment, e.g., [67, 77, 76, 66], the percentage of prior aligned entities between the KBs strongly influences the alignment accuracy. We therefore studied its impact on our approach and BootEA, evaluating both with different seed sizes, from 10 % to 50 % of the existing alignments. Results in terms of Hits@1 are reported in Fig. 5.2 for the ZH-EN and FR-EN datasets of DBP15K, similar trends being observed on the JA-EN dataset. The two methods are strongly affected by the size of the seed: when it decreases performance plummets. However, we notice that for small seed sizes, near 10 %, AlignD performs slightly better than BootEA—around 3 points better on ZH-EN and FR-EN Hits@1, 2 points on JA-EN—while for higher seed sizes the two methods are comparable. This observation tends to the robustness of AlignD to small seed sizes.

Table 5.4 – Hits@1, Hits@10 and MRR with noise-corrupted seed.

Approaches	ZH-EN			JA-EN			FR-EN		
	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR
BootEA[random(0.3),seed]	54.68	80.02	0.634	49.16	76.90	0.587	49.22	78.71	0.597
AlignD[random(0.3),seed]	55.71	80.30	0.642	51.29	78.48	0.606	51.28	80.07	0.614
RDGCN[random(0.3), seed]	61.30	76.99	–	67.17	82.69	–	82.18	92.91	–
BootEA(*glove*,300)[random(0.3),seed]	79.10	91.58	0.835	80.64	91.07	0.842	87.32	94.36	0.897
AlignD(*glove*,300)[random(0.3),seed]	79.32	91.84	0.837	80.57	91.39	0.843	87.42	94.64	0.899

Table 5.5 – t-values for the statistical significance test of A/B pairs using Hits@1. Rejection region at a risk $\alpha = 5\,\%$ for the equality of mean between A and B is $|T| > 2.262$ for the two-tail t-test.

H_0	ZH-EN	JA-EN	FR-EN
AlignD = BootEA	29.59	38.69	54.87
AlignD = RDGCN	17.51	19.65	5.69

5.3.4 Corrupted Seed Modeling

As we claimed in Sec. 5.2.2, AlignD should be more robust to the quality of the seed and should benefit, w.r.t. other methods, from an automatically extended and potentially noisy seed, due to its ability to avoid wrong alignments. We thus conduct experiments where a part of the seed is artificially corrupted, replacing a fraction of the seed alignments by randomly chosen alignments. We measure the impact of corruption on performance, corrupting 30 % of the seed. Results are reported in Tab. 5.4 where row Method[random(fraction), seed] corresponds to a given method—either AlignD, BootEA or RDGCN—trained with a randomly corrupted seed where the fraction of corrupted alignments is 0.3. Experimental results clearly indicate that AlignD is more robust w.r.t. noise in the seed than competing methods. Globally the accuracy decreases for all the methods when the seed is noisy, as expected, but AlignD still performs better than BootEA and RDGCN. This setup highlights the robustness of AlignD and encourages to extend the ground-truth seed with a prediction obtained using an off-the-shelf embedding-based entity alignment approach, such as BootEA or RDGCN. A Student test is performed on the results of the different methods over 10 runs. Tab. 5.5 gather the t-values for the tested methods, and shows the gain for AlignD over the baselines to be significant.

Table 5.6 – Hits@1, Hits@10 and MRR with an automatically extended seed.

Approaches	ZH-EN			JA-EN			FR-EN		
	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR
AlignD[BootEA]	64.66	85.10	0.715	62.82	85.60	0.704	65.94	87.15	0.732
AlignD[RDGCN]	82.20	93.97	0.863	83.84	94.07	0.874	90.38	97.03	0.926

5.3.5 Automatically Extended Seed

We also studied the impact of training AlignD with a seed predicted using an off-the-shelf embedding-based entity alignment approach. Therefore, AlignD was trained using the prediction of BootEA and RDGCN, both obtained from a ground-truth seed (containing 30 % of the alignments) at the initial iteration of these last two algorithms. The initial alignment as input to AlignD can thus be seen as the initial seed extended with BootEA or RDGCN, hence the notion of extended seed.

Results on the DBP15k datasets are given in Tab 5.6. We can see that AlignD[BootEA] (i.e., AlignD with initialization obtained from BootEA) performs better than AlignD and BootEA alone, due to the seed extension and the ability of AlignD to perform alignment with a noisy seed. We also compared AlignD[RDGCN] with RDCGN, both using Glove pre-trained embedding for entity names embedding as initialization. As previously, applying AlignD following RDGCN improves over the latter, however not being significantly better that AlignD(*glove*, 300) directly applied on the seed. We believe this is due to the high-quality of the GloVe initialization.

Globally, every extended seed experiment improves significantly the performance w.r.t. their corresponding experiment with a ground-truth seed, the best results over all our experiments being obtained with AlignD[RDGCN]. We thus conclude that using the prediction of a matching algorithm to artificially extend the size of seed leads to a more accurate final alignment with AlignD, exploiting its robustness to noise.

5.3.6 Automatically Generated Seed

Increased robustness of AlignD to errors makes it possible to overcome the need for prior alignment between KBs, replacing human-based seed alignments generation by substring matching between entities' names. Our objective here is to demonstrate the capacity of AlignD in designing a system to align KBs with no ground-truth seed at all, combining symbolic methods with embedding-based ones.

Tab. 5.7, symbolic columns, gathers results for the six symbolic methods mentioned

Table 5.7 – Hits@1 using symbolic methods alone or in combination with AlignD.

Approach	symbolic			AlignD[symbolic]		
	ZH-EN	JA-EN	FR-EN	ZH-EN	JA-EN	FR-EN
DSC	25.22	44.64	54.10	80.28	84.01	89.33
Fuzzy	35.00	44.93	46.57	71.05	81.81	84.97
Substring	24.75	44.29	57.00	70.79	76.84	85.95
Word2Vec	24.38	37.49	39.86	79.61	80.81	84.47
Levenshtein	25.07	44.27	54.05	79.95	84.08	89.2
JaroWinkler	29.06	49.01	54.73	81.21	85.30	89.61

in Sec. 5.2.3 on the DBP15k datasets. Those symbolic approaches perform better on both FR-EN and JA-EN than on ZH-EN, which is explained by the quality of the used translation model as this model appears to translate French and Japanese to English better than Chinese to English. The best symbolic approaches are substring matching for FR-EN, JaroWinkler for JA-EN and fuzzy matching for ZH-EN.

Symbolic matching of entity names is here used to provide an initial, low quality, seed for AlignD. We first compare in Tab. 5.7, last three columns, the impact of the seed generation approach on AlignD. AlignD not only improves the results of all the symbolic methods but reduces the discrepancies between them. The best Hits@1 score is obtained when combining JaroWinkler with AlignD. This is due to the fact that JaroWinkler gives more favorable rating to strings that match from the beginning —entity names—. Entity names in DBpedia, which are unique and coherent from one language to another, are thus well-matched with this method.

We further compare in Tab. 5.8 the best unsupervised system combining a symbolic approach with AlignD (AlignD[JaroWinkler]) with the state-of-the-art entity alignment methods applied on the same substring matching seed alignment. AlignD[JaroWinkler] outperformed all the methods, taking advantage of its robustness to errors in the seed, except BootEA($glove, 300$)[JaroWinkler] and AlignD($glove, 300$)[JaroWinkler]. The latter obtained the best results, outperforming the state of the art. Interestingly, AlignD without pre-trained word embeddings initialization, got a score close to the unsupervised methods using pre-trained word embeddings, which means that GloVe initialization only adds limited gain when using automatically generated seed while being hard to obtain in real world scenario. These good results are partly explained by the size of the initial alignment provided by the string matching method, which is much larger than the 30 % of the alignments used by other methods, however much noisier. We noticed the complementary

Table 5.8 – Hits@1 for alignment methods with seed automatically generated with a symbolic approach.

Approaches	ZH-EN	JA-EN	FR-EN
BootEA[JaroWinkler]	61.35	57.56	59.06
RDGCN[JaroWinkler]	65.10	75.76	83.52
AlignD[JaroWinkler]	81.21	85.30	89.61
BootEA($glove, 300$)[JaroWinkler]	82.99	84.39	91.72
AlignD($glove, 300$)[JaroWinkler]	83.12	84.58	92.15

performance of the two methods as AlignD($glove, 300$)[JaroWinkler] is better on ZH-EN and FR-EN while AlignD[JaroWinkler] is better on JA-EN, even if they do not have either the same dimension size or the same initialization. As AlignD[JaroWinkler] requires neither human intervention for seed generation nor initialization, we recommend its use rather than that of AlignD($glove, 300$)[JaroWinkler] in real-world contexts.

5.4 Conclusion

Learning-based entity alignment techniques link entities of different KBs that are the same by minimizing their distance on the *seed*, leveraging a KBs joint embedding model. We rather advocate for global entity alignment using dimension alignment. By introducing the alignment of the dimensions of the initial KB embedding spaces in the learning process, as an indirect criterion to embed similar entities together, we showed that we can limit the need for prior high-accuracy alignment. We have experimentally shown the effectiveness and robustness of our method AlignD, particularly with regard to the size and the quality of the *seed* alignment. Therefore, AlignD paves the way to a fully-automatic entity alignment system that makes no use of prior seed alignment. We thus combined AlignD using dimension alignment with a state-of-the-art entity alignment algorithm, and showed it to be not only beneficial for state-of-the-art-approaches, but also enables the substitution of the manual prior entity alignment step with an automatic alignment algorithm based on symbolic approaches of entity alignment. Globally, our alignment criterion suppresses the need for highly-accurate seed and answers a wider range of realistic scenarios where perfect handcrafted seeds are not available.

This contribution open the door towards fully exploiting the semantics of RDF knowledge bases for entity alignment. Particularly, AlignD can be designed with embedding models other than TransE including graph-based ones. It will be also interesting to au-

tomatically build prior alignment, accounting for entities attributes, rather than limiting ourselves to string matching approaches. Moreover, our alignment method, open the perspective to examine other more robust differentiable correlation coefficients instead of Pearson's correlation coefficient.

Chapter 6

Conclusion

6.1 Summary of the Contributions

This manuscript evaluates the thesis that RDF KBs are more suitable for content and KBs linking. Our goal is to capitalize as far as possible on the semantics of RDF KBs for the tasks of linking, accouting for the cardinalities of direct and indirect relations between entities. We first investigate how to define an accurate collective entity linking system that will solves the problem of linking mentions from textual content to their corresponding entities in a KB. We advocate for entity relatedness measures that will benefit for the task of collective entity linking, accouting for the semantics of the KB at hand. But, depending on the reference KB for the linking, entity relatedness measures do not bear the same semantic, and can have a very different impact on the linking. On the one hand, Wikipedia-like KBs only provide the information that two web pages share some hyperlinks, they are somehow vaguely linked. By contrast, RDF KBs define semantic relations between entities based on their use in a real word scenarios. We therefore expect RDF KBs to be more suitable for linking content and data with comparison to Wikipedia-like KBs. In order to validate our hypothesis, we inspected the entity relatedness measures used so far in the context of collective linking. The limits of these techniques are their vague semantics in the case of Wikipedia based techniques, and the under-use of KBs semantic in the case of RDF KBs. Our solution for the linking relies on the definition of several entity relatedness measures that benefit as far as possible from the semantics of RDF KBs. We first contribute with the entity relatedness measure (WSRM) which capitalizes on the semantic relations between entities in an RDF KB leveraging entities interconnections. This relatedness measure is incorporated in a lightweight collective entity linking technique which we show to be better than popular state-of-the-art collective entity linking systems and was published in the 2019 edition of the document engineering conference DocEng. The same measure was also validated within a different system that was applied to the task of processing named entities from historical newspapers (HIPE) and published in

the 2020 edition of the CEUR workshop. Moreover, and in order to highlight the interest of entity relatedness measures based on RDF KBs for the literature of entity linking, we studied the requirements for a good entity relatedness measure and proposed three measures that respect these requirements namely $ASRMP^x_m$, $x \in \{a, b, c\}$ defined in Chapter 4. $ASRMP^x_m$ allows to benefit from the semantics of RDF KBs, and introduces basic reasoning based on indirect paths which improves the linking accuracy. A collective entity linking system with the measure $ASRMP^x_m$ was published in the 2020 edition of the international semantic web conference (ISWC). The overall conclusion is that well-founded entity relatedness measures that benefit from the semantics of RDF KBs, allow improving the accuracy of the collective entity linking. The latter, ultimately facilitates the exploration of textual documents.

We investigate also how RDF KBs can be enriched accounting for public RDF KBs e.g., open linked KBs, as solution to enrich RDF KBs used for the collective entity linking. That it, we studied entity alignment techniques which aim at linking the same entities in different KBs and provide a solution for linking. The goal of entity alignment techniques is to complement KBs by interchanging the knowledge related to the entities they have in common. We proposed a global criterion that maximizes the correlation of the dimension of the multidimensional spaces where the KBs are projected. Moreover, this global criterion optimizes the distance for all the entities in the seed at once, which is known to be better than point-wise or batch-wise optimization. Our alignment technique AlignD, allows replacing the need for a ground truth seed by the predictions of symbolic approaches that match entities based on string similarity. The solution we devise to replace the need for labels in the context of entity alignment, constitutes a major advance for the literature of entity alignment as previous system were limited by the size and the quality of the seed. AlignD was published in the 2020 edition of international conference on tools with artificial intelligence (ICTAI).

Our contributions for entity linking and entity alignment are based on the use of semantic RDF KBs, and show such a KBs to be more beneficial for content linking by comparison to Wikipedia-like KBs.

6.2 Perspectives

Several possible future directions can be explored from the methods we discussed in this manuscript.

6.2.1 General Entity Relatedness Measures

The entity relatedness measures we proposed in Chapter 4 were devised for the task of collective entity linking. Nevertheless, entity relatedness measures can be used for a variety of tasks such as RDF KB visualization where the idea is to show the underlying graph for a given RDF KB; metric learning with the goal of learn the distance between two entities in general; entity retrieval where the idea is to find for a given entity, the most similar entities in a KB. For instance entity retrieval can be used in a semantic search engine. The entity relatedness measures we proposed for the collective linking can be studied further in a more general context. A thorough analysis of the proposed entity relatedness measures in the light of the requirements we defined, may benefit the above mentioned tasks.

6.2.2 Word and Entity Joint Embedding for CEL

In our solution for the CEL, we mainly used entity relatedness measures. However, embedding techniques can be used to solve all the linking problems based on one embedding space. For instance we used the skip-gram model to learn word embedding and the TransE to learn entity embedding but a joint word-entity embedding model will be more beneficial for the linking. The idea is to incorporate in one model, skip-gram properties and TransE properties and to study the quality of the final multidimensional space. We believe such an embedding space bears a strong semantic so it defines a good L2-norm for the collective entity linking and as the both entities and words are embedded in the same space, we expect mentions in text to be projected in the embedding space close to their corresponding entities from the reference KB. Entity linking problem can thus be solved with simple KNN search in the joint embedding space. Nevertheless, cosine similarity does not meet the requirement **(R3)**, we recommend to constrain the embedding space so that the cosine similarity for entities indirectly linked in the embedding do not exceed a fixed value provided by an entity relatedness measure such as the one we proposed in Chapter 4. Constrained joint-embedding space will allow building a rich semantics multidimensional space that reflects the semantics of the used entity relatedness measure, at the same time, facilitates the linking tasks. One possible solution to build this joint embedding model is to fix the embedding of the entities outputted by TransE and train the skip-gram model so that the mentions of given entity get embedded close to this entity. This joint embedding model was comparable to the skip-gram when evaluated on the analogy task [50], but we

did not extensively evaluated it, especially for the task of linking.

6.2.3 Indirect Criterion for Word and Entity Joint Embedding

The join-embedding space described above, can be learned accounting for the dimension alignment criterion described in Chapter 5. Rather than minimizing the L2-norm between entities and their mentions in the joint embedding space, this joint-embedding model can be learned by maximizing the correlation between entities and mentions globally, similarly to AlignD. Our entity alignment approach AlignD (described in Chapter 5) was designed to align entities from different KBs, the same process can be used to align an entity with its associated mentions, the basic idea is to indirectly lower the distance between an entity and its associated mentions accounting for some correlation coefficient, e.g., Pearson's correlation. While we limited ourselves in Chapter 5 to Pearson's correlation, it will be also interesting to study the properties of the correlation metrics from the literature and see which correlation coefficient works the best for a given embedding space. Particularly, one can try differentiable correlation coefficients that can be learned along with the joint embedding space. In other words to be able to learn the correlation coefficient and the KBs embedding at the same time.

6.2.4 Impact of Entity Alignment on CEL

Entity alignment provides a straightforward solution to link entities in different KBs. Moreover, entity alignment can be used to perform entity linking without going through the standard three stages pipeline. Imagine we linked a text corpus to a reference KB_1, we can align this KB_1 to a new KB, say KB_2, we thus indirectly linked the text corpus to KB_2 using the alignment process. Such a solution is not error free, therefore, it opens the question of its effectiveness. A thorough analysis of entity alignment techniques will show to what extent entity alignment is beneficial for indirect collective entity linking.

6.2.5 Local Global Entity Linking

Collective entity liking improves on entity-by-entity linking since it incorporates candidate entity interrelationships in linking process at a computationally expensive cost. We rather propose here to study the efficiency of a local collective entity linking where the global scores are computed only for mentions in a sliding window or in a sentence. The

idea is to reduce the noise introduced by the mentions that are far in a document. The local-global linking that we discuss here can be seen as a tradeoff between entity-by-entity linking and collective entity linking. Since entity-by-entity supposes the mentions in a textual document to be completly independent, while collective entity linking assumes those mentions to be stronly linked.

BIBLIOGRAPHY

[1] Ayman Alhelbawy and Robert Gaizauskas, « Graph ranking for collective named entity disambiguation », *In*: *Proceedings of the Association for Computational Linguistics*, 2014, pp. 75–80.

[2] Kazuki Ashihara et al., « Improving topic modeling through homophily for legal documents », *In*: *Applied Network Science* 5.*1* (2020), pp. 1–20.

[3] Roi Blanco, Paolo Boldi, and Andrea Marino, « Using graph distances for named-entity linking », *In*: *Science of Computer Programming* 130 (2016), pp. 24–36.

[4] Roi Blanco, Giuseppe Ottaviano, and Edgar Meij, « Fast and space-efficient entity linking for queries », *In*: *Proceedings of the ACM International Conference on Web Search and Data Mining*, 2015, pp. 179–188.

[5] Piotr Bojanowski et al., « Enriching word vectors with subword information », *In*: *Transactions of the Association for Computational Linguistics* 5 (2017), pp. 135–146.

[6] Kurt Bollacker et al., « Freebase: a collaboratively created graph database for structuring human knowledge », *In*: *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 2008, pp. 1247–1250.

[7] Antoine Bordes et al., « Translating embeddings for modeling multi-relational data », *In*: *Proceedings of the Advances in Neural Information Processing Systems*, 2013, pp. 2787–2795.

[8] Eric Brill et al., « Data-Intensive Question Answering », *In*: *Proceedings of the Text REtrieval Conference*, 2001, pp. 393–400.

[9] Razvan Bunescu and Marius Paşca, « Using encyclopedic knowledge for named entity disambiguation », *In*: *Proceedings of the European Chapter of the Association for Computational Linguistics*, 2006, pp. 9–16.

[10] Maxime Buron et al., « Ontology-Based RDF Integration of Heterogeneous Data », *In*: *International Conference on Extending Database Technology (EDBT)*, Copenhagen, Denmark, 2020, pp. 299–310.

[11] Yixin Cao et al., « Bridge Text and Knowledge by Learning Multi-Prototype Entity Mention Embedding », *In*: *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, vol. 1, 2017, pp. 1623–1633, DOI: 10.18653/v1/p17-1149, URL: http://dx.doi.org/10.18653/v1/P17-1149.

[12] Yixin Cao et al., « Multi-Channel Graph Neural Network for Entity Alignment », *In*: *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 1452–1461.

[13] Yixin Cao et al., « Neural collective entity linking », *In*: *Proceedings of the International Conference on Computational Linguistics*, 2018, pp. 675–686.

[14] Diego Ceccarelli et al., « Learning relatedness measures for entity linking », *In*: *Proceedings of the ACM International Conference on Information & Knowledge Management*, 2013, pp. 139–148.

[15] Muhao Chen et al., « Multilingual Knowledge Graph Embeddings for Cross-lingual Knowledge Alignment », *In*: *Proceedings of the International Joint Conference on Artificial Intelligence*, 2017, pp. 1511–1517.

[16] Silviu Cucerzan, « Large-scale named entity disambiguation based on Wikipedia data », *In*: *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 2007, pp. 708–716.

[17] Marcin Detyniecki, « Mathematical aggregation operators and their application to video querying », PhD thesis, Univ. Paris 6, 2000.

[18] Jacob Devlin et al., « Bert: Pre-training of deep bidirectional transformers for language understanding », *In*: *arXiv preprint arXiv:1810.04805* (2018).

[19] Mark Dredze et al., « Entity disambiguation for knowledge base population », *In*: *Proceedings of International Conference on Computational Linguistics*, 2010, pp. 277–285.

[20] Greg Durrett and Dan Klein, « A joint model for entity analysis: coreference, typing, and linking », *In*: *Transactions of the Association for Computational Linguistics* 2 (2014), pp. 477–490.

[21] Cheikh Brahim El Vaigh et al., « A correlation-based entity embedding approach for robust entity linking », *In*: *International Conference on Tools with Artificial Intelligence (ICTAI)*, VIRTUAL CONFERENCE, 2020, pp. 949–954.

[22] Cheikh Brahim El Vaigh et al., « A Novel Path-Based Entity Relatedness Measure for Efficient Collective Entity Linking », *In*: *International Semantic Web Conference*, VIRTUAL CONFERENCE, 2020, pp. 164–182.

[23] Cheikh Brahim El Vaigh et al., « IRISA System for Entity Detection and Linking at CLEF HIPE 2020 », *In*: (2020).

[24] Cheikh Brahim El Vaigh et al., « Using knowledge base semantics in context-aware entity linking », *In*: *ACM Symposium on Document Engineering 2019*, 2019, pp. 1–10.

[25] John R Firth, « A synopsis of linguistic theory, 1930-1955 », *In*: *Studies in linguistic analysis* (1957).

[26] Matthew Francis-Landau, Greg Durrett, and Dan Klein, « Capturing semantic similarity for entity linking with convolutional neural networks », *In*: *Proceedings of the North American Chapter of the Association for Computational Linguistics*, 2016, pp. 1256–1261.

[27] Octavian-Eugen Ganea and Thomas Hofmann, « Deep joint entity disambiguation with local neural attention », *In*: *Proceedings of the International Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 2619–2629.

[28] Octavian-Eugen Ganea et al., « Probabilistic bag-of-hyperlinks model for entity linking », *In*: *Proceedings of the International Conference on World Wide Web*, 2016, pp. 927–938.

[29] Amir Globerson et al., « Collective entity resolution with multi-focal attention », *In*: *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, vol. 1, 2016, pp. 621–631.

[30] Ramanathan Guha, Rob McCool, and Eric Miller, « Semantic Search », *In*: *Proceedings of the International World Wide Web Conference*, 2003, pp. 700–709.

[31] Nitish Gupta, Sameer Singh, and Dan Roth, « Entity linking via joint encoding of types, descriptions, and context », *In*: *Proceedings of the International Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 2681–2690.

[32] Zhengyan He et al., « Efficient collective entity linking with stacking », *In*: *Proceedings of the International Conference on Empirical Methods in Natural Language Processing*, 2013, pp. 426–435.

[33] Johannes Hoffart et al., « KORE », *In*: 2012, ISBN: 9781450311564, DOI: 10.1145/2396761.2396832, URL: http://dx.doi.org/10.1145/2396761.2396832.

[34] Johannes Hoffart et al., « Robust disambiguation of named entities in text », *In*: *Proceedings of the International Conference on Empirical Methods in Natural Language Processing*, 2011, pp. 782–792.

[35] Hongzhao Huang, Larry Heck, and Heng Ji, « Leveraging deep neural networks and knowledge graphs for entity disambiguation », *In*: *arXiv preprint arXiv:1504.07678* (2015).

[36] Ioana Hulpuş, Narumol Prangnawarat, and Conor Hayes, « Path-based semantic relatedness on linked data and its use to word and entity disambiguation », *In*: *Proceedings of the International Semantic Web Conference*, 2015, pp. 442–457.

[37] Heng Ji et al., « Overview of TAC-KBP2015 tri-lingual entity discovery and linking », *In*: *Proceedings of the Text Analysis Conference*, 2015.

[38] Heng Ji et al., « Overview of TAC-KBP2016 tri-lingual EDL and its impact on end-to-end cold-start KBP », *In*: *Proceedings of the Text Analysis Conference* (2016).

[39] Heng Ji et al., « Overview of TAC-KBP2017 13 languages entity discovery and linking », *In*: *Proceedings of the Text Analysis Conference*, 2017.

[40] Thorsten Joachims, « Optimizing search engines using clickthrough data », *In*: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2002, pp. 133–142.

[41] Nikolaos Kolitsas, Octavian-Eugen Ganea, and Thomas Hofmann, « End-to-end neural entity linking », *In*: *Proceedings of the International Conference on Computational Natural Language Learning*, 2018, pp. 519–529.

[42] Sayali Kulkarni et al., « Collective annotation of Wikipedia entities in web text », *In*: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009, pp. 457–466.

[43] Phong Le and Ivan Titov, « Improving entity linking by modeling latent relations between mentions », *In*: *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2018, pp. 1595–1604.

[44] Yann LeCun et al., « Gradient-based learning applied to document recognition », *In*: *Institute of Electrical and Electronics Engineers* 86.*11* (1998), pp. 2278–2324.

[45] Yankai Lin et al., « Modeling Relation Paths for Representation Learning of Knowledge Bases », *In*: *Proceedings of the International Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 705–714.

[46] Xiao Ling, Sameer Singh, and Daniel S Weld, « Design challenges for entity linking », *In*: *Transactions of the Association for Computational Linguistics* 3 (2015), pp. 315–328.

[47] Ming Liu et al., « A Multi-View–Based Collective Entity Linking Method », *In*: *ACM Transactions on Information Systems (TOIS)* 37.*2* (2019), pp. 1–29.

[48] Weiming Lu et al., « Boosting collective entity linking via type-guided semantic embedding », *In*: *Proceedings of the National CCF Conference on Natural Language Processing and Chinese Computing*, 2017, pp. 541–553.

[49] Pablo N. Mendes et al., « DBpedia spotlight: shedding light on the web of documents », *In*: *Proceedings of the International Conference on Semantic Systems*, 2011, pp. 1–8.

[50] Tomas Mikolov et al., « Distributed representations of words and phrases and their compositionality », *In*: *Proceedings of Advances in Neural Information Processing Systems*, 2013, pp. 3111–3119.

[51] George A. Miller, « WordNet: A Lexical Database for English », *In*: *Commununi-cations of the ACM* 38.*11* (1995), pp. 39–41.

[52] David Milne and Ian H Witten, « Learning to link with Wikipedia », *In*: *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2008, pp. 509–518.

[53] Jose G. Moreno et al., « Combining word and entity embeddings for entity linking », *In*: *Proceedings of European Semantic Web Conference*, 2017, pp. 337–352.

[54] Andrea Moro, Alessandro Raganato, and Roberto Navigli, « Entity linking meets word sense disambiguation: a unified approach », *In*: *Transactions of the Association for Computational Linguistics* 2 (2014), pp. 231–244.

[55] Thien Huu Nguyen et al., « Joint learning of local and global features for entity linking via neural networks », *In*: *Proceedings of the International Conference on Computational Linguistics*, 2016, pp. 2310–2320.

[56] Xiaoman Pan et al., « Unsupervised entity linking with abstract meaning representation », *In*: *Proceedings of the North American Chapter of the Association for Computational Linguistics*, 2015, pp. 1130–1139.

[57] Jeffrey Pennington, Richard Socher, and Christopher Manning, « Glove: global vectors for word representation », *In*: *Proceedings of the International Conference on Empirical Methods in Natural Language Processing*, 2014, pp. 1532–1543.

[58] Matthew Peters et al., « Deep Contextualized Word Representations », *In*: *Proceedings of the North American Chapter of the Association for Computational Linguistics*, 2018, pp. 2227–2237.

[59] Minh C. Phan et al., « Pair-linking for collective entity disambiguation: Two could be better than all », *In*: 2018.

[60] Francesco Piccinno and Paolo Ferragina, « From TagME to WAT: a new entity annotator », *In*: *Proceedings of the International Workshop on Entity Recognition & Disambiguation*, 2014, pp. 55–62.

[61] Petar Ristoski et al., « RDF2Vec: RDF graph embeddings and their applications », *In*: *Semantic Web* 10.*4* (2019), pp. 721–752.

[62] Michael Röder et al., « N^3- A collection of datasets for named entity recognition and disambiguation in the NLP interchange format », *In*: *Proceedings of International Conference on Language Resources and Evaluation*, 2014, pp. 3529–3533.

[63] Wei Shen, Jianyong Wang, and Jiawei Han, « Entity linking with a knowledge base: Issues, techniques, and solutions », *In*: *IEEE Transactions on Knowledge and Data Engineering* 27.*2* (2015), pp. 443–460.

[64] Pavel Shvaiko and Jérôme Euzenat, « Ontology Matching: State of the Art and Future Challenges », *In*: *IEEE Transactions on Knowledge and Data Engineering* 25.*1* (2013), pp. 158–176, URL: `https://hal.inria.fr/hal-00917910`.

[65] Valentin I. Spitkovsky and Angel X. Chang, « A cross-lingual dictionary for English Wikipedia concepts », *In*: *Proceedings of the International Conference on Language Resources and Evaluation*, 2012, pp. 3168–3175.

[66] Zequn Sun, Wei Hu, and Chengkai Li, « Cross-lingual entity alignment via joint attribute-preserving embedding », *In*: *Proceedings of the International Semantic Web Conference*, 2017, pp. 628–644.

[67] Zequn Sun et al., « Bootstrapping Entity Alignment with Knowledge Graph Embedding », *In*: *Proceedings of the International Joint Conference on Artificial Intelligence*, 2018, pp. 4396–4402.

[68] Yulia Tsvetkov et al., « Evaluation of word vector representations by subspace alignment », *In*: *Proceedings of the International Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 2049–2054.

[69] Ricardo Usbeck et al., « AGDISTIS - Graph-based disambiguation of named entities using linked data », *In*: *Proceedings of the International Semantic Web Conference*, 2014, pp. 457–471.

[70] Ricardo Usbeck et al., « GERBIL: General entity annotator benchmarking framework », *In*: *Proceedings of the International Conference on World Wide Web*, 2015, pp. 1133–1143.

[71] W3C, *RDF 1.1 Concepts and Abstract Syntax*, https://www.w3.org/TR/rdf11-concepts, 2014.

[72] W3C, *RDF 1.1 Primer*, https://www.w3.org/TR/rdf11-primer, 2014.

[73] W3C, *RDF 1.1 Semantics*, https://www.w3.org/TR/rdf11-mt, 2014.

[74] W3C, *RDF 1.1 Semantics*, https://www.w3.org/TR/rdf11-mt, 2014b.

[75] Han Wang et al., « Language and domain independent entity linking with quantified collective validation », *In*: *Proceedings of the International Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 695–704.

[76] Zhichun Wang et al., « Cross-lingual knowledge graph alignment via graph convolutional networks », *In*: *Proceedings of the International Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 349–357.

[77] Yuting Wu et al., « Relation-Aware Entity Alignment for Heterogeneous Knowledge Graphs », *In*: *Proceedings of the International Joint Conference on Artificial Intelligence*, 2019, pp. 5278–5284.

[78] Ikuya Yamada et al., « Joint learning of the embedding of words and entities for named entity disambiguation », *In*: *Proceedings of the International Conference on Computational Natural Language Learning*, 2016, pp. 250–259, DOI: 10.18653/v1/k16-1025, URL: http://dx.doi.org/10.18653/v1/K16-1025.

[79] Yi Yang, Ozan İrsoy, and Kazi Shefaet Rahman, « Collective Entity Disambiguation with Structured Gradient Tree Boosting », *In*: *Proceedings of the North American Chapter of the Association for Computational Linguistics*, 2018, pp. 777–786.

[80] Yuanzhe Zhang et al., « A Joint Model for Question Answering over Multiple Knowledge Bases », *In*: *Proceedings of the AAAI Conference on Artificial Intelligence*, 2016, pp. 3094–3100.

[81] Hao Zhu et al., « Iterative Entity Alignment via Joint Knowledge Embeddings », *In*: *Proceedings of the International Joint Conference on Artificial Intelligence*, 2017, pp. 4258–4264.

List of Tables

2.1 RDF triples for facts and knowledge. 23
2.2 Subset of RDF entailment rules typically used in data management 23

3.1 Statistics on the datasets used. 40
3.2 Linking accuracy (F1 score) on the TAC KBP-2017 dataset. Popularity and cosine similarity are the local mention-entity scores; S and $M_{1,2,3}$, the global features, are defined in Eq. 3.10 and Eq. 3.11 respectively. 42
3.3 Micro-averaged F1 score for different methods on the four datasets. 44
3.4 F1 score for VINCULUM (as reported in [46]) and WSRM on the AIDA datasets. 44
3.5 t-values for the statistical significance test of A/B pairs using the micro-averaged F1 scores. Rejection region at a risk $\alpha = 5\,\%$ for the equality of mean between A and B is $T > 2.539$ for the one-tail t-test and $|T| > 2.093$ for the two-tail t-test. 44

4.1 Entity relatedness measures in the light of well-foundedness requirements: $\times$ indicates the requirement is met, while $\sim$ indicates it is only partially met. 51
4.2 ASRMP^x_m in the light of well-foundedness requirements: $\times$ indicates the requirement is met, while $\sim$ indicates it is only partially met. 55
4.3 F1 scores for various classifiers within the entity linking system for TAC-KBP. 61
4.4 Linking F1 score on the TAC-KBP2017 dataset. Popularity and cosine similarity are the local mention-entity scores; the sum (S_m) and max ($M^{(k)}_m$) global features are defined in Eq. 3.10 and Eq. 3.11 resp. 62
4.5 Time in (min.) for different entity relatedness measures. 64
4.6 Micro-averaged F1 score for different collective entity linking systems on four standard datasets. 64

5.1 Statistics on datasets of DBP15k and DWY100K. 79

5.2 Hits@1, Hits@10 and MRR with ground-truth seed on DBP15K datasets. The top half results are taken from the literature while we produced bottom ones by running new experiments. 80
5.3 Hits@1, Hits@10 and MRR with ground-truth seed on DWY100K datasets. 81
5.4 Hits@1, Hits@10 and MRR with noise-corrupted seed. 83
5.5 t-values for the statistical significance test of A/B pairs using Hits@1. Rejection region at a risk $\alpha = 5\,\%$ for the equality of mean between A and B is $|T| > 2.262$ for the two-tail t-test. 83
5.6 Hits@1, Hits@10 and MRR with an automatically extended seed. 84
5.7 Hits@1 using symbolic methods alone or in combination with AlignD. . . . 85
5.8 Hits@1 for alignment methods with seed automatically generated with a symbolic approach. 86

List of Figures

3.1 Illustration of a mention-entity graph within a document: the two mentions (Apple and Steve Jobs) are linked to candidate entities with some local score (solid lines) while the entity relatedness appears as weighted dotted lines. In this toy example, the strong relation between the entities Steve Jobs and Apple Inc helps in jointly selecting those two. 30
3.2 Distribution of entity pairs (Y-axis) per number of relations between entities (X-axis) in the BaseKB RDF KB. 34

4.1 Sample paths between candidate entities of entity mentions from TAC-KBP2017 dataset. Two entities are connected either with a solid line if their WSRM value is above 0.01, or with a dotted line if their non null WSRM value is below 0.01 48
4.2 Example of aggregation with ASRMP^a_m 53
4.3 Example of aggregation with ASRMP^b_m 53
4.4 Example of aggregation with ASRMP^c_m 54
4.5 Linking F1 score for various aggregation strategies. 61

5.1 Pipeline used to automatically compose a seed. 77
5.2 Hits@1 on entity alignment w.r.t. the seed size, expressed as the proportion of alignments used as seed. 82

List of Publications

The list of publications in decreasing chronological order:

— Kazuki Ashihara, Cheikh Brahim El Vaigh, Chenhui Chu, Benjamin Renoust, Noriko Okubo, Noriko Takemura, Yuta Nakashima and Hajime Nagahara. Improving topic modeling through homophily for legal documents. *Applied Network Science*, volume 5, pages 1–20, 2020

— Cheikh Brahim El Vaigh, Guillaume Le Noé-Bienvenu , Guillaume Gravier and Pascale Sébillot. IRISA System for Entity Detection and Linking at CLEF HIPE 2020. In *Proceedings of the CEUR Workshop*, 2020

— Cheikh Brahim El Vaigh,François, Robin Allesiardo, Guillaume Gravier and Pascale Sébillot. A correlation-based entity embedding approach for robust entity linking. In *Proceedings of the International Conference on Tools with Artificial Intelligence*, 2020

— Cheikh Brahim El Vaigh, François Goasdoué, Guillaume Gravier and Pascale Sébillot. A Novel Path-based Entity Relatedness Measure for Efficient Collective Entity Linking. In *Proceedings of the International Semantic Web Conference*, 2020

— Cheikh Brahim El Vaigh, François Goasdoué, Guillaume Gravier and Pascale Sébillot. Using knowledge base semantics in context-aware entity linking. In *Profeceeding of ACM Symposium on Document Engineering*, pages 1–10, 2019

UNIVERSITE BRETAGNE LOIRE / MATHSTIC

UNIVERSITÉ DE RENNES 1

Titre : Utilisation de connaissances ontologiques dans la liaison de contenus et de données appliquée au journalisme de données.

Mot clés : traitement automatique des langues, bases de connaissances ontologiques, liage d'entités, mesure de relations entre entités, alignement d'entités

Résumé : Cette thèse s'intéresse à la création de liens entre contenus textuels et bases de connaissances ontologiques (BC). Elle fait appel à plusieurs domaines de recherche : le traitement automatique des langues, la recherche d'information et le web sémantique, notamment l'utilisation de BC fondées sur le modèle RDF. Nous proposons d'une part d'étudier le liage d'entités collectif qui cherche à relier simultanément les mentions d'entités présentes dans un texte aux entités d'une BC. Notre contribution porte sur la définition de mesures sémantiques bien fondées qui exploitent les propriétés des BC pour améliorer l'état de l'art, et permettent d'introduire du raisonnement. D'autre part, nous nous intéressons à l'alignement de différentes BC, moyennant des approches de plongement des bases dans des espaces de grandes dimensions. Cet alignement permet l'enrichissement des BC, et indirectement l'amélioration du liage d'entités collectif. Pour ce faire, nous proposons un nouveau critère qui se fonde sur l'alignement des dimensions des espaces de plongement des BC, et permet de résister à un alignement a priori bruité entre les BC, voire de supprimer ce besoin d'alignement manuel.

Title: Content and data linking leveraging ontological knowledge in data journalism

Keywords: Natural language processing, ontological knowledge bases, entity linking, entity relatedness, entity alignment

Abstract: This thesis is about the creation of links between textual content and ontological knowledge bases (KBs). It pertains several areas of research: natural language processing, information retrieval and semantic web, and in particular RDF-based KBs. We propose to study collective entity linking, which consists in linking at once mentions of entities present in a textual document to entities in a KB. To that end, we leverage semantic measures, i.e., entity relatedness measures which exploit the relationships between the entities in a KB. We contribute by the definition of well-founded entity relatedness measures that benefit to the extent possible from the properties of RDF KBs through (basic) reasoning, and thus allow to improve the state-of-the-art. Furthermore, we are also interested in the alignment of different KBs, based on KBs embedding techniques. This alignment not only allows to enrich the KBs at hand, but also to indirectly improve the collective entity linking. We contribute by an alignment criterion, based on the alignment of the dimensions of the KBs embedding spaces, which, notably does not need any prior knowledge to perform said KBs alignment.