Using Background Knowledge to Enhance Biomedical Ontology Matching

Amina Annane

THESIS TO OBTAIN THE DEGREE OF DOCTOR OF PHILOSOPHY (PH. D.)
FROM THE UNIVERSITY OF MONTPELLIER

In computer science

Prepared at:
Information, Structure, Systems (I2S) graduate school
Laboratory of Informatics, Robotics and Microelectronics of Montpellier (LIRMM), France
In partnership with
Higher School of Computer Science (ESI), Algeria

Using Background Knowledge to Enhance Biomedical Ontology Matching

Defended by Amina ANNANE
On October 29th, 2018

Under the supervision of Zohra BELLAHSENE
and Faiçal AZOUAOU

Reviewers

Patrick Lambrix	Pr.	Linköping University
Karima Akli-Astouati	HDR	University of science and technology Houari boumediene

In front of a jury composed by

Yacine Chalal	Pr.	Higher School of Computer Science	President
Mokrane Bouzeghoub	Pr.	University of Versailles Saint-Quentin	Examiner
Souhila Kaci	Pr.	University of Montpellier	Examiner
Zohra Bellahsene	Pr.	University of Montpellier	Director
Faiçal Azouaou	HDR	Higher School of Computer Science	Director
Clement Jonquet	Dr.	University of Montpellier	Guest

UNIVERSITÉ DE MONTPELLIER

ECOLE NATIONALE SUPÉRIEURE D'INFORMATIQUE

Abstract

Life sciences produce a huge amount of data (e.g., clinical trials, scientific articles) so that integrating and analyzing all the datasets related to a given research question like the correlation between phenotypes and genotypes, is a key element for knowledge discovery. The life sciences community adopted Semantic Web technologies to achieve data integration and interoperability, especially ontologies which are the key technology to represent and share the increasing amount of data on the Web. Indeed, ontologies provide a common domain vocabulary for humans, and formal entity definitions for machines.

A large number of biomedical ontologies and terminologies has been developed to represent and annotate various datasets. However, datasets represented with different overlapping ontologies are not interoperable. It is therefore crucial to establish correspondences between the ontologies used; an active area of research known as ontology matching.

Original ontology matching methods usually exploit the lexical and structural content of the ontologies to align. These methods are less effective when the ontologies to align are lexically heterogeneous i.e., when equivalent concepts are described with different labels. To overcome this issue, the ontology matching community has turned to the use of external background knowledge resources (BK) as a semantic bridge between the ontologies to align. This approach arises several new issues mainly: (1) the selection of these background resources, (2) the exploitation of the selected resources to enhance the matching results. Several works have dealt with these issues jointly or separately. In our thesis, we made a systematic review and historical evaluation comparison of state-of-the-art approaches.

Ontologies, others than the ones to align, are the most used background knowledge resources. Related works often select a set of complete ontologies as background knowledge, even if, only fragments of the selected ontologies are actually effective for discovering new mappings. We propose a novel BK-based ontology matching approach that selects and builds a knowledge resource with just the right concepts chosen from a set of ontologies. The conducted experiments showed that our BK selection approach improves efficiency without loss of effectiveness.

Exploiting background knowledge resources in ontology matching is a double-edged sword: while it may increase recall (i.e., retrieve more correct mappings), it may lower precision (i.e., produce more incorrect mappings). We propose two methods to select the most relevant mappings from the candidate ones: (1) based on a

set of rules and (2) with Supervised Machine Learning. We experiment and evaluate our approach in the biomedical domain, thanks to the profusion of knowledge resources in biomedicine (ontologies, terminologies and existing alignments).

We evaluated our approach with extensive experiments on two Ontology Alignment Evaluation Initiative (OAEI) benchmarks. Our results confirm the effectiveness and efficiency of our approach and overcome or compete with state-of-the-art matchers exploiting background knowledge resources.

As a preamble to the main subject of the thesis, we have reconciled, in a semi-automatic way, multilingual mappings between French and English ontologies. We have described the multilingual-mappings produced with semantic properties, and made them available to the scientific community as a structured resource stored on the SIFR BioPortal ontology repository.

Résumé

Les sciences de la vie produisent de grandes masses de données (par exemple, des essais cliniques et des articles scientifiques). L'intégration et l'analyse des différentes bases de données liées à la même question de recherche, par exemple la corrélation entre phénotypes et génotypes, sont essentielles pour découvrir de nouvelles connaissances. Pour cela, la communauté des sciences de la vie a adopté les techniques du Web sémantique pour réaliser l'intégration et l'interopérabilité des données, en particulier les ontologies. En effet, les ontologies représentent la brique de base pour représenter et partager la quantité croissante de données sur le Web. Elles fournissent un vocabulaire commun pour les humains, et des définitions d'entités formelles pour les machines.

Un grand nombre d'ontologies et de terminologies biomédicales a été développé pour représenter et annoter les différentes bases de données existantes. Cependant, celles qui sont représentées avec différentes ontologies qui se chevauchent, c'est à dire qui ont des parties communes, ne sont pas interopérables. Il est donc crucial d'établir des correspondances entre les différentes ontologies utilisées, ce qui est un domaine de recherche actif connu sous le nom d'alignement d'ontologies.

Les premières méthodes d'alignement d'ontologies exploitaient principalement le contenu lexical et structurel des ontologies à aligner. Ces méthodes sont moins efficaces lorsque les ontologies à aligner sont fortement hétérogènes lexicalement, c'est à dire lorsque des concepts équivalents sont décrits avec des labels différents. Pour pallier à ce problème, la communauté d'alignement d'ontologies s'est tournée vers l'utilisation de ressources de connaissance externes en tant que pont sémantique entre les ontologies à aligner. Cette approche soulève plusieurs nouvelles questions de recherche, notamment : (1) la sélection des ressources de connaissance à utiliser, (2) l'exploitation des ressources sélectionnées pour améliorer le résultat d'alignement. Plusieurs travaux de recherche ont traité ces problèmes conjointement ou séparément. Dans notre thèse, nous avons fait une revue systématique et une comparaison des méthodes proposées dans la littérature. Puis, nous nous sommes intéréssés aux deux questions.

Les ontologies, autres que celles à aligner, sont les ressources de connaissance externes (Background Knowledge : BK) les plus utilisées. Les travaux apparentés sélectionnent souvent un ensemble d'ontologies complètes en tant que BK même si, seuls des fragments des ontologies sélectionnées sont réellement efficaces pour découvrir de nouvelles correspondances. Nous proposons une nouvelle approche qui sélec-

tionne et construit une ressource de connaissance à partir d'un ensemble d'ontologies. La ressource construite, d'une taille réduite, améliore, comme nous le démontrons, l'efficience et l'efficacité du processus d'alignement basé sur l'exploitation de BK.

L'exploitation de BK dans l'alignement d'ontologies est une épée à double tranchant : bien qu'elle puisse augmenter le rappel (i.e., aider à trouver plus de correspondances correctes), elle peut réduire la précision (i.e., générer plus de correspondances incorrectes). Afin de faire face à ce problème, nous proposons deux méthodes pour sélectionner les correspondances les plus pertinentes parmi les candidates qui se basent sur : (1) un ensemble de règles et (2) l'apprentissage automatique supervisé. Nous avons expérimenté et évalué notre approche dans le domaine biomédical, grâce à la profusion de ressources de connaissances en biomédecine (ontologies, terminologies et alignements existants).

Nous avons effectué des expériences intensives sur deux benchmarks de référence de la campagne d'évaluation de l'alignement d'ontologie (OAEI) réalisée chaque année dans la communauté pour évaluer les outils. Nos résultats confirment l'efficacité et l'efficience de notre approche et dépassent ou rivalisent avec les meilleurs résultats obtenus par les meilleurs systèmes d'alignement exploitant des ressources de connaissance externes.

En préambule du sujet principal de la thèse nous avons réconcilié, d'une manière semi-automatique, des correspondances multilingues entre des ontologies françaises et anglaises. Les correspondances produites ont été décrites à l'aide des propriétés sémantiques et mises au service de la communauté scientifique sous forme de ressource structurée stockée dans l'entrepôt d'ontologies biomédicales SIFR BioPortal.

Contents

List of Figures

List of Tables

Chapter

1 Introduction

Contents

1.1 Context and motivations

In life sciences, such as medicine, biology, genetics, etc., researchers produce and manage a large number of biomedical datasets (e.g., clinical trials, scientific articles). Integrating and analyzing all the datasets related to a given research question, like the correlation between genotype and phenotype (Coulet et al., 2008), is a key element for knowledge discovery (Collins et al., 2003). Therefore, researchers are increasingly publishing these datasets on the web to make them available for further studies (see Figure 1.1).[1]

However, an effective exploitation of the knowledge included in these datasets raises several challenges about their representation, integration and interoperability. These challenges are the same addressed by the semantic web community on a larger scale to manage the increasing amount of data published on the web in various domains.

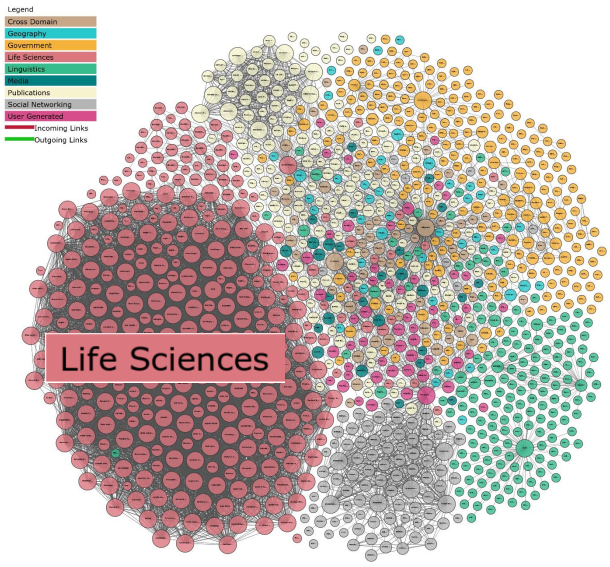

Figure 1.1: Linking Open Data cloud diagram 2017.

The semantic web is an extension of the current web that aims at giving a meaning to published data such that both humans and machines can semantically process and exploit them (Berners-Lee et al., 2001). Indeed, an enormous set of interlinked documents in natural language is published on the web. These documents have been created to be exploited by humans. However, with the exponential growth of data on the web, manually analyzing and integrating web document knowledge is becoming increasingly difficult. Therefore, the semantic web community develops

[1] `https://lod-cloud.net/`

models and techniques to enable machines managing web document knowledge.

Ontologies are the semantic web's key technology for representing, sharing and querying web data (Berners-Lee et al., 2001). An ontology may be seen as a conceptual model that represents the knowledge of a given domain. An ontology is composed of a set of concepts, a set of semantic relations that link these concepts and a set of axioms (i.e., logic rules) which ensure the coherence of the model. Ontologies have multiple applications (Hoehndorf et al., 2015). An ontology can be used as a query model for datasets or as a basis for their integration (Lambrix and Tan, 2006). Further, it can be used to annotate different datasets composing a given repository and serves as an index. The annotation process consists in identifying the ontology concepts in the content of the annotated datasets (Tchechmedjiev et al., 2018).

Due to the decentralized nature of the semantic web, a large number of ontologies has been developed during the last decade. To manage these ontologies and facilitate their reuse, several ontology repositories (or libraries) have been created (d'Aquin and Noy, 2012). This is especially true for the biomedical domain which has its own ontology repositories such as the NCBO BioPortal (Noy et al., 2009), the OBO Foundry (Smith et al., 2007) and the Ontology Lookup Service (Côté et al., 2008). The number of biomedical ontologies is continuously increasing, for instance, in the NCBO BioPortal, the number of ontologies has increased from 134 to more than 700 over a period of nine years (see Figure 1.2).[2]

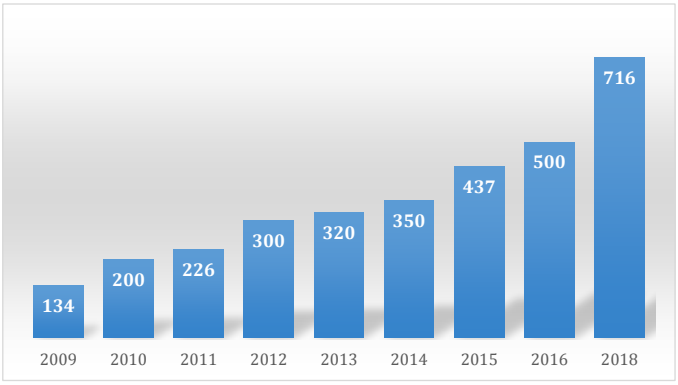

Figure 1.2: Number of ontologies per year in the NCBO BioPortal repository.

Creating a large number of ontologies in the same domain increases the overlapping between these ontologies. For instance, all the ontologies showed in Figure 1.3, have a fragment that describes human diseases.[3] This is problematic for data integration. Indeed, datasets related to the same research problem, which are annotated

[2]We have collected these statistics from publications related to the NCBO BioPortal repository.

[3]On the figure, we showed the ontology acronyms used in NCBO BioPortal to reference ontologies.

or represented with different ontologies are not interoperable. One possible solution to address this issue is to establish correspondences (or mappings) between the semantically related entities of ontologies belonging to the same domain; this process is known as *ontology matching*.

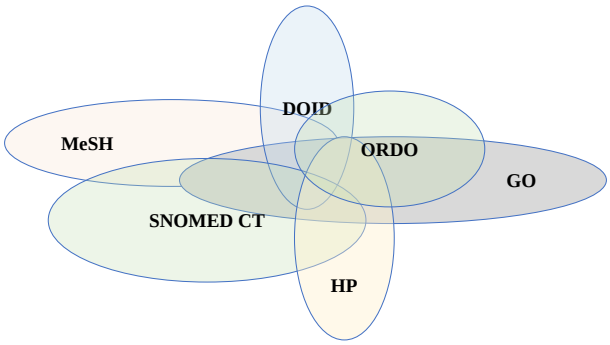

Figure 1.3: Biomedical ontologies with overlapping fragments.

To illustrate our statements, we present an example in Figure 1.4. Let us suppose that we have several datasets related to genotype-disease correlation studies, which are annotated with different ontologies such as the human disease ontology (DOID) and the orphanet rare disease ontology (ORDO).

To seek the same information in the different datasets, one has to customize a query for each ontology, i.e., a query for DOID, a query for ORDO, etc. However, when having mappings between ontologies, only a single query, customized for one ontology (DOID in our example), is required.

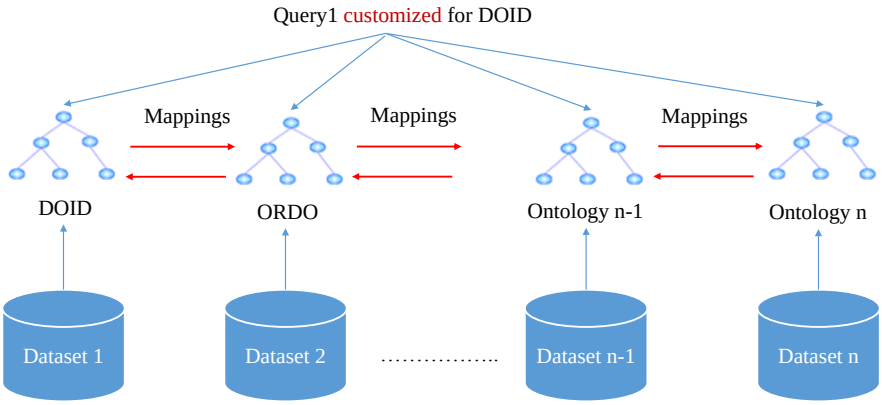

Figure 1.4: Example: querying information using mappings.

Ontology matching is an active area of research because of its wide range of applications such as ontology engineering, data integration, information sharing, etc. (Euzenat and Shvaiko, 2013). However, ontologies are highly heterogeneous

(Klein, 2001, Halevy, 2005) because they have been designed independently, by different developers, following diverse modeling principles and patterns. Furthermore, the diversity of their heterogeneity: syntactic, lexical and structural, as well as their size and formats makes ontology matching a complex and challenging task.

1.2 Challenges

Original automatic ontology matching methods are mainly based on the lexical and structural information of the ontologies to align; this is known as *direct matching*. To that end, several similarity measures have been developed (Cheatham and Hitzler, 2013, Ngo et al., 2013). The effectiveness of these approaches depends on the lexical and structural overlap of the input ontologies. Indeed, they fail to discover mappings when the input ontologies have equivalent concepts described by different labels (no common labels, no similar labels), and they are structured according to different modeling views (Aleksovski et al., 2006a, Pesquita et al., 2013).

To overcome this semantic heterogeneity, the community has turned to the exploitation of external knowledge resource(s), commonly called background knowledge resources. In contrast to direct matching, this approach is known as *indirect matching* or *BK-based matching*, as it exploits external resources to identify mappings between the ontologies to align. The type of these knowledge resources span from thesaurus, lexical resources, ontologies other than those to align, etc. BK-based ontology matching approach proved a successful alternative (Sabou et al., 2008, Locoro et al., 2014). Indeed, the empirical results show that exploiting external knowledge resources improves the alignment quality, especially by increasing recall i.e., by finding mappings that were missed by direct matching methods (Aleksovski et al., 2006a, Mascardi et al., 2010, Annane et al., 2016a). BK-based matching arises several new research issues mainly:

1.2.1 Background knowledge resource selection

Exploiting knowledge resources as a semantic bridge between the ontologies to align helps to find mappings missed by the direct matching, thus increasing the quality of produced alignments. However, the success of this matching approach depends on the quality of the knowledge resources exploited in the matching process. Hence, the challenge is to select effective background knowledge resources for a given ontology matching task (Shvaiko and Euzenat, 2013). In the initial related works, the selection of these resources was performed manually. Manual resource selection assumes that users, who have to make the selection, are familiar with all the available knowledge resources, which is not always possible, especially in domains having a large number of knowledge resources such as biomedicine. Therefore, an *automatic*

selection method is required to increase the applicability of the BK-based ontology matching approach (Faria et al., 2014). Furthermore, some knowledge resources are of large size (e.g., DBpedia). In most cases, for a given ontology matching task, we argue that only fragments from these resources are actually effective. Using the whole resources affects heavily the efficiency of the matching process. Thus, another challenge for the automatic selection of knowledge resources is the extraction of the *effective fragments* from the large knowledge resources.

1.2.2 Background knowledge resource exploitation

Exploiting background knowledge resources in the matching process includes three steps. The first one, called anchoring, aims at linking the entities of the ontologies to align to the entities of the selected resources. This is usually performed by a direct matching between the ontologies to align and the selected resources. The second one, called derivation, deduces (or derive) semantic relations between the anchored entities – entities of the ontologies to be aligned – according to the relations linking the anchors in the background knowledge resources (see Figure 1.5). Finally, the third step aggregates the derived mappings and selects the most relevant ones. These three steps are common to all BK-based matching methods, however each one provides several choices:

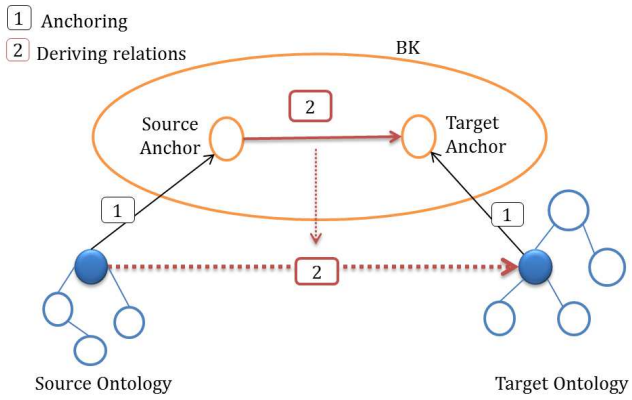

Figure 1.5: Exploiting a background knowledge resource to generate mappings.

- The direct matcher to use for the anchoring step: a simple syntactic matcher or a sophisticated one.
- The entities to be considered in the BK-based matching: all entities of the ontologies to align or only entities that have not been directly mapped. Indeed, BK-based matching aims to improve and complement the direct matching alignments.

- To derive mappings, the selected knowledge resources may be exploited independently of each other, or combined together.

Different choices lead to different configurations and results. Determining the configuration that ensures the best performance is challenging. In addition, exploiting background knowledge resources in ontology matching is a double-edged sword: while it may increase recall (i.e., retrieve more correct mappings), it may lower precision (i.e., generate more incorrect mappings) (Shvaiko and Euzenat, 2013). Consequently, *selecting* correct mappings among the candidate ones in the context of BK-based matching is particularly challenging.

1.3 Research contributions

Before proceeding to present the contributions of our research in detail, we clarify the main goal of this dissertation, which is to provide a generic, efficient and effective approach to enhance the direct matching results using ontologies as external knowledge resources. Indeed, in the literature, ontologies are the most used knowledge resources. This is explained by the fact that they are *structured* resources, and their knowledge is *validated* by the community. Hence, it is easier to exploit them as background knowledge than to use other knowledge resource types such as textual documents. Research questions that we address in this thesis can be stated as follows:

Research questions. *Given a repository of knowledge resources (i.e., set of ontologies), and an ontology matching task (i.e., two ontologies to align):*

- *How to select effective and efficient background knowledge resources for the matching task from the initial repository?*
- *How to effectively exploit the selected background knowledge resources to enhance the direct-matching alignment?*

We attempted to answer these research questions by making the following contributions:

1. Review of methods related to the selection and exploitation of background-knowledge resources in ontology matching

Since the exploitation of knowledge resources proved a successful technique to improve the direct matching results, several works have investigated the benefit of this approach. Hence, the necessity of studying and comparing the related works. We reviewed the different methods dealing with the two main steps of BK-based ontology matching that are the selection and exploitation of background knowledge

resources. In addition, we provided a synthetic classification of the studied methods. Finally, we presented a comparative evaluation of the BK-based ontology matching systems by analyzing their performance results obtained during Ontology Alignment Evaluation Initiative (OAEI) 2012-2016 campaigns. We thus evaluate the benefit of exploiting background knowledge resources and the improvement achieved with systems implementing this approach comparing to the systems that do not.

2. A novel efficient and effective background knowledge selection/building method

In the literature, several works have dealt with the automatic selection of ontologies as background knowledge for ontology matching (see Chapter 4). All the proposed methods consist in selecting m ontologies from the n ones available with $m <= n$. These methods return a set of *complete* ontologies, however we believe that within each selected ontology, especially large ones, only small fragments may actually prove effective. Hence, the issue is that of the selection of these ontology fragments and their combination to build an effective and efficient background knowledge resource.

At the best of our knowledge, in the context of ontology matching, our work is the first that promotes the extraction of effective fragments from background knowledge resources and their combination.

In our thesis, we tackle this issue by selecting only the relevant concepts from the background knowledge ontologies related to the matching task. We then combine these selected concepts with mappings to build a new knowledge resource, that we called *the built BK*. As it is shown in Chapter 4, the built BK has a small size comparing to that of the background knowledge ontologies, which improves the efficiency of the BK-based matching. In addition, in Chapter 5, we show that the reduced size of the built BK does not affect its effectiveness.

The built BK interconnects concepts from different background knowledge ontologies enabling deriving mappings through several intermediate ontologies.

3. Methods for exploiting the selected/built background knowledge resource

As we explained previously in Section 1.2.2, exploiting background knowledge resources in ontology matching includes three steps. Our contributions concerns the second (i.e., deriving candidate mappings) and the third (i.e., aggregating and selecting the most relevant mappings) steps. We experimentally compared several derivation strategies according to the combination of intermediate resources, and the entities to consider in the BK-based matching. We showed that the combination generates more correct mappings (see Section 5.3.1), and considering all the entities in the indirect matching provides better results but consumes more time (see Section 6.5). Moreover, to improve the efficiency of the derivation process, we implemented an algorithm that reduces the number of the returned paths (see

Section 6.4).

The built BK is a graph where nodes are ontology concepts, and edges are equivalence mappings. The derivation process returns paths of different length between the entities of the ontologies to align. To select the final mappings, we had to compose the mappings composing each path, and aggregating the different paths representing the same candidate mappings. First, we proposed to use the multiplication function to compose mapping scores, and aggregating with the maximum function i.e., for each candidate mapping, keeping the occurrence with the highest score. We then defined a set of rules to select the most relevant mappings. Secondly, we assumed that the mapping selection method would be more effective when having a deeper description of each candidate mapping. Hence, we designed a set of 27 selection attributes for each candidate mapping e.g., number of paths, average path length. Manually assessing the performance of all the possible combinations of these 27 selection attributes is not feasible. Therefore, we proposed to transform the problem of mapping selection into a classification problem and use a machine learning algorithm to combine these selection attributes, and classify candidate mappings. We implemented the two selection methods and experimentally compared their performance.

4. GBM: a prototype of a Generic BK-based ontology Matcher

Existing BK-based matchers implement the indirect matching technique in their internal architectures, which makes any adaptation or reuse of the code difficult. Hence, when someone attempts to improve a particular step in the BK-based matching process, he will have to code the whole process from scratch, which was our case. Therefore, we judged interesting to propose to the community a Generic BK-based Matcher (GBM).

GBM implements all the contributions mentioned above, and may be reused with any existing matcher. In addition, it takes as input a set of various parameters related to different BK-based matching steps, which makes it customizable and very appropriate to perform evaluation experiments. GBM has participated, with YAM++ as a direct matcher, in the OAEI 2017, and OAEI 2017.5 campaigns, where it has been successful on the biomedical benchmarks, and top ranked in several tasks.

5. Multilingual mapping reconciliation

As a preliminary work of this thesis, we constructed a resource composed of multilingual mappings, which may be reused as a background knowledge resource to match multilingual ontologies. Indeed, following a semi-automatized workflow, we reconciled more than 228K mappings between ten English ontologies hosted on NCBO BioPortal and their French translations on SIFR BioPortal (Jonquet et al., 2016). We have formalized and represented the generated mappings with semantic properties, and stored them on SIFR BioPortal in RDF format to be accessible to the

community. Reconciling the mappings turned more complex than expected because the translations are rarely exactly the same as the original ontologies.

This contribution is related to *multilingual* ontology matching, while all the other contributions of this thesis are related to *monolingual* ontology matching. Therefore, we decided to present the multilingual mapping reconciliation chapter as an appendix to this thesis (see Appendix A).

1.4 Outline of the dissertation

Figure 1.6 shows the organization of the remaining chapters of this dissertation. Each chapter references its related papers that have been published within this thesis. The list of these papers is presented in the next Section (Section 1.5).

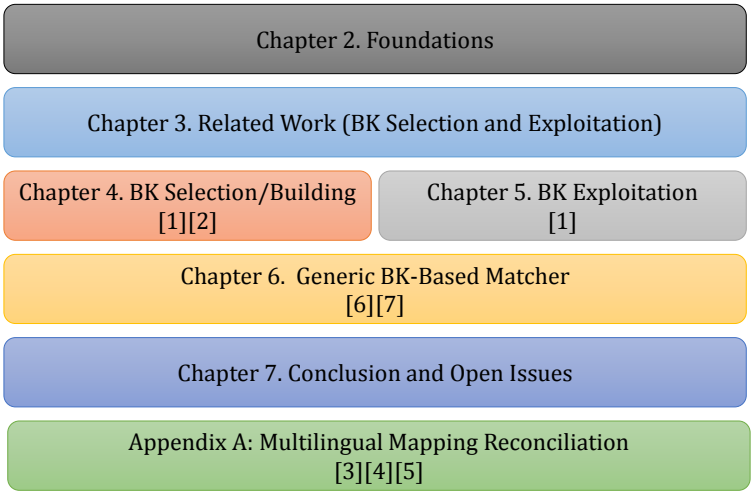

Figure 1.6: Organization of the remaining of this dissertation.

- *Chapter 2* provides the definitions of the main concepts used in this thesis. In particular, it introduces the necessary preliminaries on ontology matching and BK-based ontology matching. The second part of this chapter presents the general workflow of BK-based ontology matching, and explains its sub-tasks.

- *Chapter 3* reviews and compares the various works related to the exploitation of background knowledge resources in ontology matching. It includes three main sections. The first one focuses in methods dealing with the automatic selection of background knowledge resources, while the second section describes the methods exploiting the selected resources in the matching process. Finally, the third section provides an evaluation of the BK-based matching approaches based on the OAEI results.

- *Chapter 4* starts by presenting the novel method that we have proposed to select (or build) a background knowledge resource, called **built BK**, from a set of initial ontologies. It then presents the experimental materials used, explains the adopted implementation techniques, and discusses the evaluation results regarding the size and the efficiency of the built BK.
- *Chapter 5* describes our methods dealing with the exploitation of the built BK in the matching process. More precisely, this chapter explains in detail (i) how we may exploit the built BK to derive candidate mappings, and (ii) how to select the most relevant mappings among the candidate ones.
- *Chapter 6* describes GBM, a **G**eneric **B**K based **M**atcher, which implements our BK selection and exploitation methods. GBM provides a set of parameters that allows various configurations. In this chapter, we present and explain these different parameters. Moreover, we present the results of our participation in OAEI 2017 and OAEI 2017.5 campaigns.
- *Chapter 7* summarizes the thesis with a discussion about the different results and concludes with some perspectives for future research directions.
- *Appendix A* describes our work related to the reconciliation of multilingual mappings between French and English ontologies. First, we present the treated ontologies, the workflow followed, and we discuss the choice of the semantic properties used to represent the multilingual mappings produced. Secondly, we show and discuss the results of this study.

1.5 Publications

The following published papers are partial outputs of this thesis.

International journals

[1] **Amina Annane**, Zohra Bellahsene, Faiçal Azouaou, and Clement Jonquet, Building an effective and efficient background knowledge resource to enhance ontology matching, *Journal of web Semantics*, 2018. `https://doi.org/10.1016/j.websem.2018.04.001`

International conferences

[2] **Amina Annane**, Zohra Bellahsene, Faiçal Azouaou, and Clement Jonquet, Selection and combination of heterogeneous mappings to enhance biomedical ontology matching, in *20th International Conference on Knowledge Engineering and Knowledge Management*, EKAW, Bologna, Italy, November 2016, vol. 10024 LNAI, pp. 19–33.

[3] **Amina Annane**, Vincent Emonet, Faical Azouaou, and Clement Jonquet, Multilingual mapping reconciliation between English-French biomedical ontologies, in *6th International Conference on web Intelligence, Mining and Semantics*, WIMS, Nîmes, France, June 2016, pp. 13:1-13:12.

French conferences

[4] **Amina Annane**, Vincent Emonet, Faiçal Azouaou, and Clement Jonquet, Réconciliation d'alignements multilingues dans BioPortal, in *27th Journées francophones d'Ingénierie des Connaissances*, IC, Montpellier, France, June 2016.

[5] Clement Jonquet, **Amina Annane**, Khedidja Bouarech, Vincent Emonet, and Soumia Melzi, SIFR BioPortal : Un portail ouvert et générique d'ontologies et de terminologies biomédicales francaises au service de l'annotation sémantique, in *16th Journées Francophones d'Informatique Médicale*, JFIM, Geneve, Switzerland, June 2016.

Workshops

[6] **Amina Annane**, Zohra Bellahsene, Faiçal Azouaou, and Clement Jonquet, YAM-BIO: Results for OAEI 2017 (System paper), in *12th International Workshop on Ontology Matching*, OM, Vienna, Austria, October 2017, pp. 201–206.

[7] Ernesto Jimenez-Ruiz, Tzanina Saveta, Ondřej Zamazal, Sven Hertling, Michael Röder, Irini Fundulaki, Axel-Cyrille Ngonga Ngomo, Mohamed Ahmed Sherif, **Amina Annane**, Zohra Bellahsene, Sadok Ben Yahia, Gayo Diallo, Daniel Faria, Marouen Kachroudi, Abderrahmane Khiat, Patrick Lambrix, Huanyu Li, Maximilian Mackeprang, Majid Mohammadi, Maciej Rybinski, Booma Sowkarthiga Balasubramani and Cassia Trojahn, Introducing the HOBBIT platform into the Ontology Alignment Evaluation Campaign, in *13th International Workshop on Ontology Matching*, OM, Mountery, USA, October 2018.

Chapter

2 Foundations

Contents

2.1 Introduction

Our work focuses on BK-based ontology matching, which is a branch of the ontology matching problem. In this chapter, we introduce the preliminaries required for the readability and understanding of our thesis manuscript. In the first part of this chapter, we start by defining the basic concepts of the ontology matching problem, the evaluation measures, and what we mean by background knowledge in the context of ontology matching (Section 2.2). The second part is dedicated to the description of the general workflow of BK-based ontology matching (Section 2.3). Indeed, based on the related works, we proposed a general workflow which consists of two main components: (i) the selection of the background knowledge resources, (ii) the exploitation of the selected resources in the matching process to derive mappings between the ontologies to be aligned.

2.2 Definitions

2.2.1 Ontology

In philosophy, ontology is the study of the nature of Being and the essence of things. In the early 1990s computer scientists, particularly those in Artificial Intelligence, gave to the term a new, but related, meaning. In the literature, there are several definitions of ontology, the most quoted one is proposed by Gruber (Gruber, 1995): *an ontology is a formal, explicit specification of a shared conceptualization.* This definition identifies four main concepts involved: an abstract model of a phenomenon termed *conceptualization*, a precise mathematical description hints the word *formal*, the precision of concepts and their relationships clearly defined are expressed by the term *explicit*, and the existence of an agreement between ontology users is hinted by the term *shared* (O'Leary, 2005, Foguem et al., 2008). An ontology O is mainly composed of a set of concepts C, a set of relations R among these concepts and a set of axioms A:

$$O = (C, R, A)$$

Concept is a class of things grouped together due to some shared property. It is named with a label, called preferred label, and sometimes with additional information such as alternative names (synonyms).

Axioms are used to formalize domain knowledge and make constraints on ontology entities. We may cite disjointness, equivalence, restriction or cardinality axioms for concepts, and transitivity, symmetry, functional or inverse axioms for properties.

When R contains single relation, the is-a relation, O belongs to a specific type of ontologies, called taxonomy, which is the most widely used type of ontologies (Ivanova and Lambrix, 2013), particularly in the biomedical domain. In the following, we use *ontologies* and *taxonomies* interchangeably.

2.2.2 Ontology matching

The following definitions were adopted from (Euzenat and Shvaiko, 2007, Euzenat and Shvaiko, 2013).

A **Similarity measure** is a function $f : E_s \times E_t \rightarrow [0..1]$ where E_s is the set of O_s entities and E_t is the set of O_t entities. For each pair of entities (e_s, e_t), a similarity measure computes a real number, generally between 0 and 1, expressing the similarity between the two entities by comparing their syntactic or structural information (Cheatham and Hitzler, 2013,Ngo et al., 2013). Other similarity measures may use external resources such as WordNet to compute the similarity between the two entities (Pedersen et al., 2004).

A **Mapping** (or a correspondence) between an entity e_s (e.g., concept, relation) belonging to ontology O_s and an entity e_t belonging to ontology O_t is a four-tuple of the form: $m = \langle e_s, e_t, r, k \rangle$ where:

- r is a relation between e_s and e_t such as equivalence ($\equiv$), subsumes ($\sqsupseteq$), subsumed by ($\sqsubseteq$), etc.
- k is a confidence score (typically in the $[0, 1]$ range) holding for the correspondence between the entities e_s and e_t. The k value is the score returned by one similarity measure, or a combination of several ones.

This thesis focuses on biomedical ontologies, which are mainly taxonomies as discussed in Section 2.2.1. Hence, we have interested in finding mappings only between concepts.

An **Alignment** of ontologies O_s and O_t is a set of mappings (or correspondences) between their entities (concepts and properties).

Ontology matching can be formally defined as a function that takes two ontologies O_s and O_t, a set of parameters P, and a set of resources R, and returns an alignment A between O_s and O_t.

A **Matcher** is an algorithm that implements one similarity measure or combines several to discover mappings between the input ontologies. In addition, a matcher includes a decision function to select which mappings will be kept in the produced alignment (Duchateau and Bellahsene, 2016). For instance the decision function may be based on a threshold value: only mappings that have a score equal to or superior than the threshold value are kept in the produced alignment. In the following, we refer to a matcher by the letter M, and we adopt the following annotation:

$$M(O_s, O_t) = A = \{m_1, m_2, ..., m_n\}.$$

Where A represents the alignment generated by the matcher M between the ontologies O_s and O_t. A is composed of n mappings m_i

Figure 2.1 shows an example extracted from (Ivanova and Lambrix, 2013). There are two fragments of two ontologies. The orange nodes are concepts of the mouse ontology[1], while the blue nodes are concepts of the NCIT ontology.[2] As we can see, the two fragments includes anatomy concepts. Directed edges represent the is-a relations between concepts, and the bidirectional dashed edges are mappings between the ontology concepts. These mappings establish links between the mouse anatomy and human anatomy.

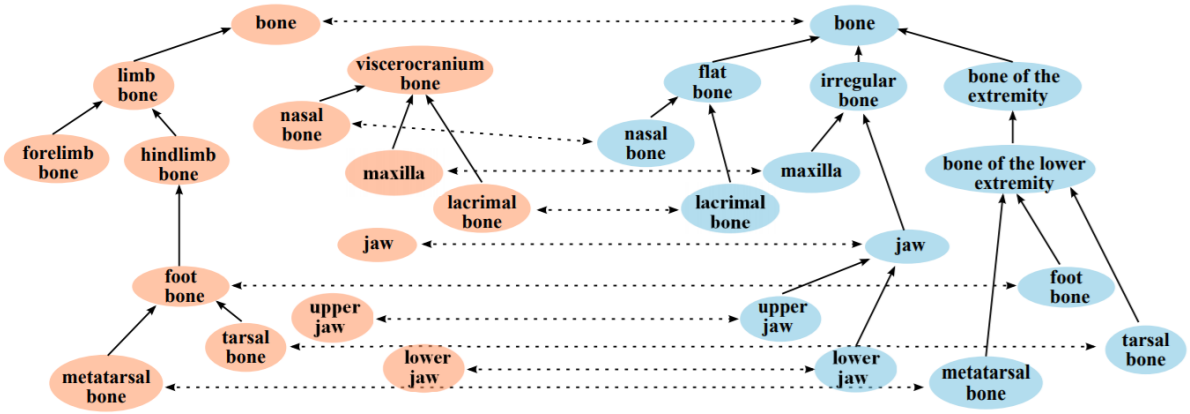

Figure 2.1: Example: ontology matching.

2.2.3 Ontology matching evaluation

Evaluating alignments produced by a given matcher is a challenging issue. Currently, the evaluation is performed using the three measures: precision, recall and F-measure (Do et al., 2002). These measures are computed against a reference alignment that contains all the correct mappings. Precision is defined as the number of correctly identified mappings divided by the total number of mappings found (correct + incorrect). Recall is defined as the number of correctly identified mappings divided by the number of all possible correct mappings (the size of the reference alignment). A perfect precision score of 1.0 means that every mapping returned by the matcher is correct; precision measures correctness. A perfect recall score of 1.0 means that all correct mappings were returned; recall measures completeness. The F-measure is the harmonic mean of precision and recall. It measures the overall accuracy of an alignment.

Let A be an alignment produced by a given matcher and R the reference alignment. Precision, Recall and F-measure are computed as follows:

[1] https://bioportal.bioontology.org/ontologies/MA
[2] https://bioportal.bioontology.org/ontologies/NCIT/

$$Precision = \frac{|A \cap R|}{|A|}$$

$$Recall = \frac{|A \cap R|}{|R|}$$

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall}$$

2.2.4 Background knowledge

In the context of ontology matching, there is no commonly accepted or strict definition of what background knowledge is. We define it as any set of external knowledge resources that provides lexical or semantic information about the domain(s) of the ontologies to align or some of the entities therein. It could be any datasets related to the ontologies to align, other ontologies than the ones to align, other previously generated mappings, lexical sources, the Web, etc.

In this manuscript, we use the acronym **BK** to refer to a non empty set of background knowledge resources used within the matching process. For instance, if such a resource is an ontology, we will call it a *BK ontology*. Similarly, the expression *BK-based method* denotes a method that exploits a set of background knowledge resources within the matching process.

2.2.5 Direct matching vs. BK-based matching

Initial ontology matching methods were based only on the exploitation of the lexical and structural content of the ontologies to be aligned, which is known as *direct matching* or *content-based matching*. However, direct matching is less effective when the ontologies to align use different labels to describe equivalent concepts, or they are structured according to different modeling views (Aleksovski et al., 2006a,Pesquita et al., 2013).

To overcome this semantic heterogeneity, the community has turned to the exploitation of external knowledge resource(s), commonly called background knowledge resources. In contrast to direct matching, this approach is known as *indirect matching*, *BK-based matching* or *context-based matching* (Locoro et al., 2014), as it exploits external resources to identify mappings between the ontologies to align. We note that the objective of BK-based matching is to complement direct matching but not to replace it. Indeed, direct matching may identify mappings that are missed with the BK-based matching and vice-versa.

Figure 2.2 shows a realistic example in the context of life-sciences, originally presented in (Aleksovski et al., 2006b). When directly matching the ontology CRISP to the ontology MeSH, no relation is found between the two concepts CRISP:Brain and MeSH:Head. Indeed, no syntactic similarity between the concept labels. In addition, MeSH contains the concept Brain but it is classified under the concept Central nervous system, which is no way related to the concept Head (different modeling

views). To overcome this semantic heterogeneity, we may exploit an external knowledge resource, the FMA ontology in our example. The concept CRISP:Brain is anchored to the concept BK:Brain and the concept MeSH:Head is anchored to the concept BK:Head. In the BK, the concepts BK:Brain and BK:Head are related via the relation *is part of*. Hence, we may derive the mapping CRISP:Brain *is part of* MeSH:Head.

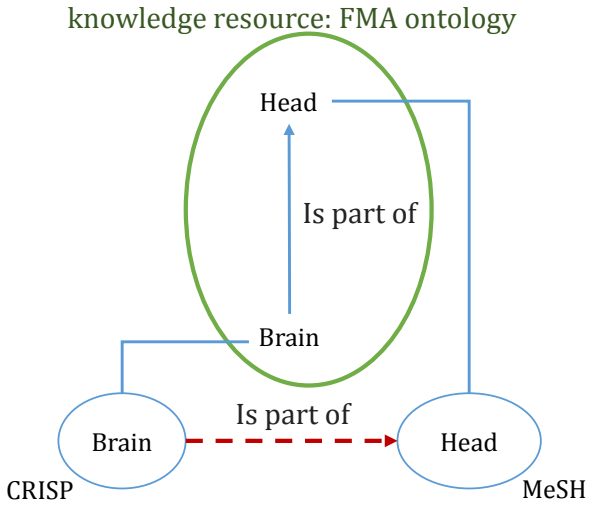

Figure 2.2: Example of BK-based matching.

2.2.6 Ontology Alignment Evaluation Initiative

Organizing and evaluating the growing number of ontology alignment systems (or methods) needs united rules and organization. The Ontology Alignment Evaluation Initiative (OAEI) is a coordinated international initiative to fill this need (Amini, 2016). It has held an annual evaluation of ontology alignment systems since 2004.[3] The main goals of the OAEI are:

- assessing strengths and weaknesses of matching systems;
- comparing performance of techniques;
- increase communication among algorithm developers;
- improve evaluation techniques;
- most of all, helping improving the work on ontology matching.

The OAEI has different tracks such as Anatomy, Conference, MultiFarm, etc. The results of the participating systems are published for further analysis.

[3] `http://oaei.ontologymatching.org/`

2.2.7 Supervised machine learning

Supervised machine learning is the task of automatically inferring a function from training data (Mehryar et al., 2012). The learned function $f : x \rightarrow y$ maps the input object x to an output y. When using the machine learning technique for a classification task, the learned function is called a *classifier*. The input x is composed of a set of attribute values that describe the object to classify, while the output y is the class in which the object x will be classified by the learned classifier. The training data is a set of objects already classified (containing both attributes and class), while the test data is the set of objects to classify. A supervised machine learning algorithm analyzes the training data and produces a classifier that will be used to classify the test data objects.

2.3 Common BK-based ontology matching workflow

In related work, there were two propositions of the general workflow of BK-based matching. The first one included only the BK exploitation step with the two sub-steps: anchoring and derivation. At that time the issue of BK selection was not highlighted neither represented (Aleksovski et al., 2006b, Safar et al., 2007). The second one described in (Locoro et al., 2014) is richer. However, the authors have focused only on ontological resources as BK. In addition, in their workflow the BK selection and BK exploitation steps are based on the anchoring step –called contextualization step– while some works did not use anchoring for BK selection (Quix et al., 2011, Chen et al., 2014). In the following we propose a generic BK-based ontology-matching workflow that covers most existing works. It includes two main steps: (1) BK Selection and (2) BK exploitation (see Figure 2.3).

2.3.1 Knowledge resource pool

It is a set of knowledge resources, $KRP = \{KR_1, KR_2, .., KR_n\}$, from which the BK, $BK = \{KR_1, KR_2, .., KR_m\}$, will be selected ($BK \subseteq KRP$). It may include all ontologies on the web (Sabou et al., 2008), a local repository of ontologies (Faria et al., 2014), existing mappings (Annane et al., 2016a), lexical resources or any combination of the previous ones.

Choosing the initial set of knowledge resources– the knowledge resource pool – may be seen as a preselection of the BK to be used in the matching process. A very large set of knowledge resources, such as all knowledge resources on the web, is time consuming and may require a lot of computational resources in the BK selection process, and a reduced set may eliminates effective knowledge resources. Hence, identifying the knowledge resource pool is an important task that should be well thought. Currently, this task is always performed manually.

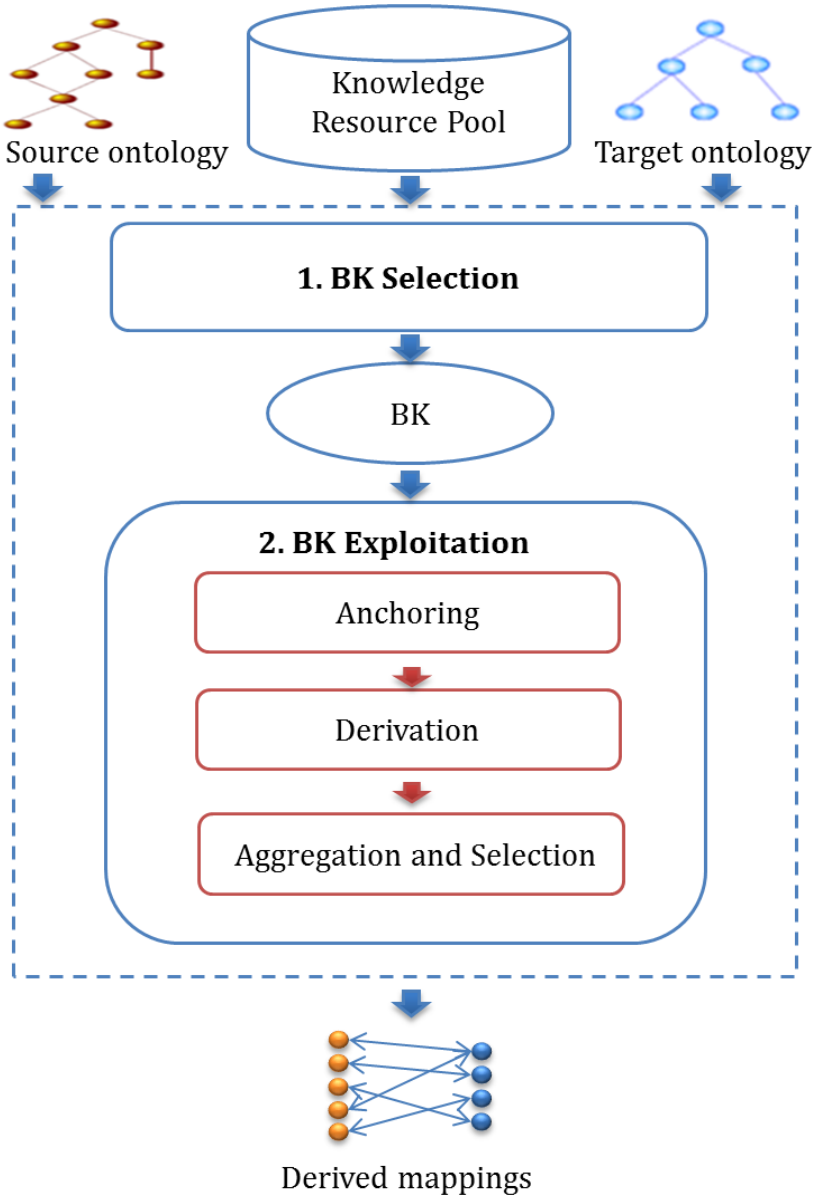

Figure 2.3: General workflow of BK-based ontology matching.

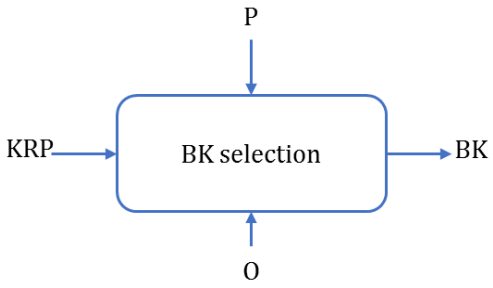

Figure 2.4: BK Selection.

2.3.2 BK selection

Effective background knowledge resources, to be used in the matching process, are those containing knowledge beyond that contained in the ontologies to match but which is relevant to match them.

BK selection is the process that attempts to find such relevant resources from a given knowledge resource pool. However, there is no measure that allows evaluating the effectiveness of a given BK before exploiting it. Indeed, currently, the evaluation is based on the comparison between the alignments obtained with and without a given BK. If the evaluation measures, especially Recall, with BK are higher than those without, we conclude that the BK used is effective.

Definition: We can formally define BK selection as a function that takes as input the knowledge resource pool KRP, ontologies to be aligned O, and optionally, a set of parameters P (e.g., threshold values), and returns the BK to be used in the matching process (see Figure 2.4).

Basically, it is about selecting a set of knowledge resources from the knowledge resource pool (e.g., one or more ontologies from a repository). Indeed, related works demonstrate that not all resources contribute with the same degree to the enhancement of the matching quality (Faria et al., 2014,Hartung et al., 2012). Furthermore, it may happen that using several resources will provide the same result as using one resource, but will require more computational resources . Therefore, several methods have recently been proposed to automatically select the BK from a given knowledge resource pool (Hartung et al., 2012, Faria et al., 2014, Quix et al., 2011). Another alternative is to dynamically build a customized BK by combining the resources of the knowledge resource pool in one global resource, then select the appropriate fragment from it. For example, in our previous work (Annane et al., 2016a), when using existing mappings as a knowledge resource pool, we selected only the mappings that were relevant for the ontology matching task avoiding the burden and complexity of dealing with whole ontologies as a BK. Another example can be found in (Kingkaew, 2012), where the author has built a customized BK resource from the textual content of a set of web pages.

2.3.3 BK exploitation

Exploiting the selected BK in the matching process includes three steps. The first one, called *anchoring*, aims at linking the entities of the ontologies to align to the entities of the selected resources. The second one, called *derivation*, deduces semantic relationships between the anchored entities – entities of the ontologies to be aligned – according to the relations linking the BK entities. Finally, the third step *aggregates* the candidate mappings derived and *selects* the most relevant ones to produce the alignment.

2.3.3.1 Anchoring

Anchoring, called also Contextualization in (Locoro et al., 2014), is a direct matching between the ontologies to be aligned and the selected BK. The aim of anchoring is to localize the entities of the ontologies to be aligned in the BK (Aleksovski et al., 2006b, Sabou et al., 2008). Thus, the BK may serve to generate mappings for the localized entities.

Definition: Let M be a matcher, O_s and O_t two ontologies to align, e_s an entity belonging to O_s, e_t an entity belonging to O_t and e'_s, e'_t entities belonging to the BK. Anchoring consists in producing two alignments A_s and A_t with the matcher M, where:

- $A_s = M(O_s, BK)$ a set of mappings of the form $m = \langle e_s, e'_s, r, k \rangle$
- $A_t = M(BK, O_t)$ a set of mappings of the form $m = \langle e'_t, e_t, r, k \rangle$

BK entities e'_s and e'_t that appear in A_s and A_t are called *anchors*, while e_s and e_t are called the *anchored entities*.

Depending on the used matcher, the relationship r may be equivalence only or including other types (e.g., subsumes and disjoint relationships). In principle, any matcher may be used for anchoring; in practice, the matcher used is usually a fast matcher (Locoro et al., 2014).

Anchoring is an important step that heavily impacts the final matching results. Indeed, the BK can not be exploited to match the non-anchored entities.

2.3.3.2 Candidate mapping derivation

Candidate mapping derivation is mainly based on the anchors resulted from the previous step.

Definition: Candidate mapping derivation consists in finding the relations linking the anchors (i.e., e'_s and e'_t) in the BK when these relations exist. Then, deriving (inferring) the relations between the anchored entities (i.e., e_s and e_t) by composing the relations of the three mappings as follows:

- $m_1 = \langle e_s, e'_s, r_1, k \rangle$,

- $m_2 = \langle e'_t, e_t, r_2, k \rangle$,
- $m_3 = \langle e'_s, e'_t, r_3, k \rangle$.

The first two mappings are resulted from the anchoring step, while the third one is inferred from the structure of the BK used (see Figure 2.2). The term *structure* refers to the semantic relations linking the BK entities. The third mapping may be the composition of several relations in the BK used.

We note that mappings generated with the indirect matching approach are sometimes represented as triples ($m = \langle e_s, e_t, r \rangle$) instead of four-tuple (Aleksovski et al., 2006a, Annane et al., 2016a). Indeed, in these cases the derivation of the semantic relation between the source and target entities is performed by exploiting the BK structure and not with similarity measures. Consequently, there is no score for this kind of mappings.

The exploration of the BK structure to derive mappings has several configurations depending on: (i) The relationships to produce in the final alignment; (ii) The type of semantic relationships to consider in the exploration (is-a, part-of, etc.); (iii) The number of resources that composes the BK (one or several); and (iv) The type of the background knowledge resources (existing mappings, text, ontologies, etc.). These different options have resulted in various derivation techniques reported in the literature that we will discuss in the next chapter.

2.3.3.3 Candidate mapping aggregation and selection

The result of the derivation step is a set of *candidate mappings*. Aggregating the derived candidate mappings and selecting the most relevant ones is a classical task in ontology matching process. In the direct matching approach, all candidate mappings have a score which is the combination of the similarity measures used. Then, usually, a threshold value is computed statically or dynamically to decide whether to keep or not a given candidate mapping in the final alignment. However, similarity scores are sometimes absent in the mappings produced by some BK-based methods (Aleksovski et al., 2006a, Sabou et al., 2008, Annane et al., 2016a). Furthermore, depending on the quality and diversity of the BK used, candidate mappings are not always correct (Tordai et al., 2010). Therefore, there is a need of using alternative strategies, especially in case of multi-resources background knowledge. Indeed, for the same couple (e_s, e_t), it is possible to have anchors in more than one background knowledge resource, then derive different semantic relationships from one resource to another (Sabou et al., 2008).

Definition: Let $BK = \{KR_1, KR_2, ..., KR_m\}$ be the BK used to derive a mapping between two entities e_s and e_t, $m_i = \langle e_s, e_t, r_i, k_i \rangle$ a derived mapping between e_s and e_t using the knowledge resource KR_i with $i \leq m$.

The aggregation is the strategy that combines the different mappings m_i into one mapping $\langle e_s, e_t, r, k \rangle$ where r is the combination of different r_i and k is the

combination of different k_i, while the selection is the strategy that allows deciding to keep or not the derived mapping in the final alignment.

2.4 Conclusion

In the first part of this chapter, we have introduced the main notions of the ontology matching field, which are necessary for understanding our manuscript. In the second part, we have presented the general workflow of the BK-based ontology matching approach and described its different tasks. This workflow will serve as a common denominator to describe and compare the related works in the next chapter.

Chapter

3 Related Work

Contents

3.1 Introduction

BK-based ontology matching is an approach that exploits external knowledge resources to overcome the semantic heterogeneity between the ontologies to be aligned. In the last decade, several works have investigated the use of this approach to enhance the ontology matching results. In this chapter we review and compare these works trying to get answers to the following research questions:

- In which cases the use of BK is justified and necessary?
- What are the application domains in which BK-based matching approach can be used?
- What is the cost of using the BK-based matching approach?
- BK-based ontology matching an alternative or a complementary solution?

The review includes three main sections. In the first section, we present the different BK selection methods. Then, in the second section, we discuss the various BK exploitation methods according to a synthetic classification that we have elaborated. In the third section, we compare and analyze the results of BK-based matching systems, which are obtained within Ontology Alignment Evaluation Initiative (OAEI) 2012-2016 campaigns. We thus evaluate the benefit of exploiting BK and the improvement achieved by this approach with regard to the systems that do not use BK. Finally, we conclude with some considerations in response to the questions arisen above.

3.2 Review of automatic BK selection methods

BK selection is a critical step in the BK-based matching since it determines the BK to be used in the matching process. Initial works exploiting external resources in ontology matching, the BK selection step was performed manually. Although the effectiveness of the manual selection, it is not practical, especially when having a large knowledge resource pool. Indeed, manual BK selection assumes the person doing the selection has an expertise and a deep understanding of the available knowledge resources, which is not always possible. To overcome this limitation, several works have investigated the automating of the BK selection process. In the following, we will review these works according to the classification showed in Figure 3.1.

3.2.1 BK selection from the Web

Using Search Engines

To the best of our knowledge, the first work that has dealt with the automatic BK selection was presented in (Sabou et al., 2008). The authors have considered

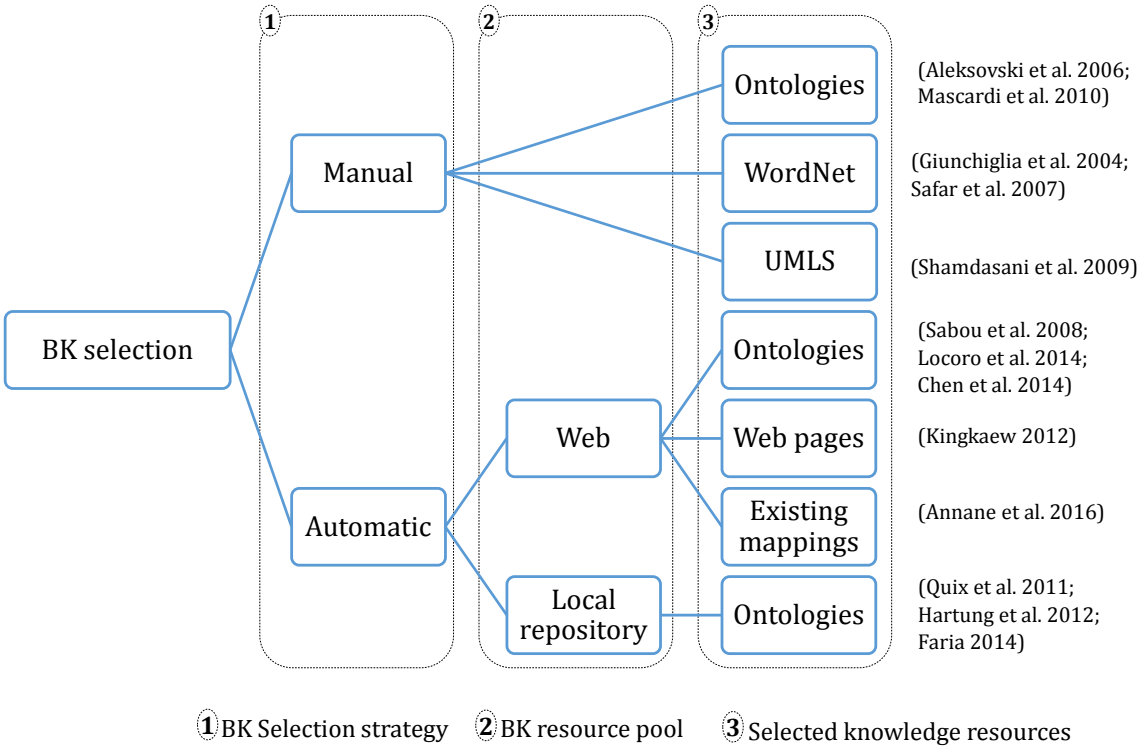

Figure 3.1: Classification of BK selection methods.

all ontologies indexed by the semantic search engine *Swoogle* (Ding et al., 2004) as a knowledge resource pool. For each candidate mapping (source and target concepts), one or several ontologies are selected dynamically as background knowledge. The authors experimented various selection strategies. According to the adopted strategy, for each candidate mapping, selected BK ontology should cover one or both concept labels. To query Swoogle the concept labels were used as they are (i.e., without any modifications), or undergone syntactic and lexical pretreatments (e.g., string normalization, synonyms with WordNet). With the normalized labels, the experimental results were better because they allowed to retrieve BK ontologies that have been missed when searching with the original labels.

Performing a BK selection for each candidate mapping is time consuming. Therefore, in (Locoro et al., 2014) the authors proposed to select a BK–a set of ontologies– once for the whole matching task, and not for each candidate mapping as in the previous work. They used the semantic search engine Watson (d'Aquin et al., 2007) to search for ontologies. The selection was controlled by the number of anchors that have to be found between the ontologies to be matched and candidate BK ontologies.

In addition to ontologies, there was an attempt that to use web pages (i.e., their textual content) as BK. The text represents a rich knowledge resource, but it is more difficult to exploit than ontologies. It should be processed, generally using natural language processing techniques to produce a structured knowledge resource that will serve as a BK. In (Kingkaew, 2012), the authors proposed to compose a query dynamically using the important terms in the ontologies to be aligned (i.e., the most frequent terms). The retrieved web pages were downloaded to a local database and indexed using Apache Lucene, then they were ranked using the similarity measure proposed in (Quix et al., 2011) and described below in Section 3.2.2. Finally, the top-k web pages are selected.

Using web services of an ontology repository In (Chen et al., 2014), the authors have considered the NCBO BioPortal as a knowledge resource pool for matching biomedical ontologies. NCBO BioPortal is a rich repository of biomedical ontologies and datasets (more than 600 ontologies) (Noy et al., 2009) that offers various web services such text annotation, ontology recommendation, etc. The selection method includes two main steps. The first one produces a baseline alignment, called A, between the ontologies to be aligned with a lexical matcher. The second step extracts labels of the concepts belonging to the alignment A, then for each extracted label, a query is made and sent to the NCBO BioPortal to search for all ontologies that have a concept with this label. Other statistics are returned such as the number of label synonyms in each candidate BK ontology. The selection process stops if the number of the candidate BK ontologies does not change after a specified number N of queries to the repository or when there are no more labels to check. Finally, the five top ontologies are selected according to the number of labels and synonyms found.

3.2.2 BK selection from a local repository

Instead of using simply the number of anchors as a selection criteria, (Quix et al., 2011), (Hartung et al., 2012) and (Faria et al., 2014) have proposed more sophisticated measures to select a set of BK ontologies from a local repository. In the following, we will describe these measures. The notations O_S, O_T and O_{BK} refer to ontology source, ontology target and candidate BK ontology respectively.

Similarity measure. Using information retrieval techniques, (Quix et al., 2011) proposed to index each ontology of the knowledge resource pool (a local repository of ontologies) as a document with the vector space model (Salton et al., 1975) by using Apache Lucene. The indexing process takes into account the lexical information of the ontology (labels, comments, etc.) but also the structural information: several structural features have been computed and aggregated into a boosting factor. This factor is used to prioritize ontologies with a high class hierarchy since they allow to infer more relationships. The authors did not provide detail about how they have computed and aggregated the structural features.

The selection of the BK ontologies consists of two steps. In the first step, two queries are dynamically constructed –one for each ontology source and target– using their lexical information. Using the index built previously, each query returns many candidate BK ontologies. If no BK ontology is found in the local repository, another query is made to search ontologies on the web. The retrieved ontologies are indexed and added to the local repository. The second step consists in ranking the candidate BK ontologies, and selecting the top-k ones. For that end, the authors proposed the following similarity measure:

$$\alpha(sim(O_{BK}, O_S) + sim(O_{BK}, O_T)) - \beta|sim(O_{BK}, O_S) - sim(O_{BK}, O_T)|.$$

The idea is to prioritize the BK ontologies that are the most similar to the source and target ontologies (i.e., maximize $sim(O_{BK}, O_S)+sim(O_{BK}, O_T)$) and are similar to both ontologies, not only to one (i.e., minimize $sim(O_{BK}, O_S) - sim(O_{BK}, O_T)$). The $sim(O_{BK}, O_x)$ function returned the Lucene similarity score between the query made of O_x and the indexed ontology O_{BK}.

Effectiveness measure. (Hartung et al., 2012) proposed a measure to rate the effectiveness of a BK ontology for a given ontology matching task. The authors proposed to anchor the ontologies to be aligned to all ontologies in the knowledge resource pool –a local repository– using a given matcher M. For each BK ontology, the anchoring produces two alignments: $A_{S,BK} = M(O_S, O_{BK})$, and $A_{BK,T} = M(O_{BK}, O_T)$, which allows to compute its effectiveness score with the following formula:

$$eff(O_S, O_{BK}, O_T) = \frac{2|range(A_{S,BK}) \cap domain(A_{BK,T})|}{|O_S|+|O_T|}.$$

Where $range(A_{S,BK})$ and $domain(A_{BK,T})$ are the O_{BK} concepts that have mappings in A.

This measure is based on the number of the BK ontology concepts that are mapped to both source and target concepts. Indeed, these BK concepts will play the role of intermediate and generate mappings between the ontologies to be aligned.

Using the effectiveness score, the authors proposed two BK selection strategies, called *topKByEffectiveness* and *topKByComplement*. As their names suggest, the first one consists simply in ranking all candidate ontologies using the effectiveness scores and selecting the top-k ones. The second one takes into account the complementarity: the BK ontology with the highest effectiveness score is the first one selected, then the second one is the most effective according to the parts of source and target ontologies that are not covered by the first BK ontology and so on. According to the experimental results, the second strategy provided the best results.

Mapping gain measure. A similar work has been presented in (Faria et al., 2014), where a selection measure, called *mapping gain*, is designed to select the most effective BK ontologies. The measure computes the number of mappings in an alignment A that is generated by exploiting a given BK with respect to another alignment B.

$$MG(A, B) = \frac{|A \cap \neg B|}{|B|}$$

The BK selection process is subdivided into two steps: a ranking step and a selection step. The first one allows to identify and rank the candidate BK ontologies. For each ontology in the knowledge resource pool, a mapping gain score is computed with respect to the direct alignment of the ontologies to be aligned. In this step, BK ontologies that have a mapping gain score less than a defined threshold are eliminated. The second step reevaluates the preselected BK ontologies taking into account the complementarity to select those to be used in the matching process. The first BK ontology selected is the one that has the highest mapping gain score comparing to the baseline alignment, then for selecting the second BK resource, the B alignment is the baseline enriched with the new mappings identified thanks to the first BK ontology and so on.

3.2.3 Discussion

Ontologies are the knowledge resources the most used as BK in ontology matching. Indeed, ontologies are structured knowledge resources validated by the community and can be directly exploited, while unstructured resources (e.g., web pages) require more treatments to structure their knowledge before exploiting them. Hence, almost all automatic BK selection methods deal with the selection of a set of ontologies as background knowledge among the candidate ones. However, the selection of other

knowledge resource types (e.g., datasets published on the web) may be an interesting research issue in the future.

The most of BK selection methods described previously are based on anchoring source and target ontologies on all knowledge resource pool, which is time consuming, especially for large knowledge resource pool. The idea proposed in (Quix et al., 2011) is more efficient: it uses information retrieval techniques to preselect a reduced set of candidate BK ontologies, then applies a selection measure only on the preselected ontologies.

According to the experimental results presented in the reviewed methods, it is not always possible to find one knowledge resource that fills the semantic gap between the ontologies to be aligned. Hence, it is more effective to select a multiple resource BK. In addition to the overlap between the ontologies to be aligned and the BK ontologies, (Hartung et al., 2012) and (Faria et al., 2014) have taken into account the complementarity criteria to avoid the redundancy and ensure the most efficient BK.

The mapping gain is the unique measure that performs a selection after deriving mappings using each BK ontology. Indeed, the measure takes as parameters two alignments between the ontologies to be aligned: the first with a BK and the second without this BK.

Currently, there is no measure or an approach to evaluate the effectiveness of a given BK selection method separately from the BK exploitation method. The evaluation is done at the end of the matching process by comparing the two alignments: (i) the one generated without exploiting the selected BK and (ii) the one that did. If the second one is better, we conclude that the selected BK is effective.

3.3 Review of BK exploitation methods

In this section, we review the various methods used to exploit background knowledge resources in ontology matching. We present and compare the reviewed works according to the BK exploitation tasks: (i) anchoring, (ii) derivation, and (iii) aggregation and selection.

3.3.1 Anchoring

As explained in Chapter 2, anchoring is a direct matching between the ontologies to be aligned and the selected BK. It aims to find BK entities related to the entities of the source and target ontologies, which permits to exploit the BK as a mediator. Usually, it is performed with an automatic matcher, except in (Aleksovski et al., 2006a) where the anchoring was manual (by a human expert) and automatic.

In the literature, the complexity of the matcher used for the anchoring step varies from one work to another. It was a simple matcher that implements only a token-based string equality in (Locoro et al., 2014), a combination of label inclusion

and Levenstein similarity measures in (Aleksovski et al., 2006a), a trigram similarity measure in (Groß et al., 2011) and a combination of syntactic and lexical similarity measures in (Mascardi et al., 2010). In other works, more sophisticated matchers have been used for anchoring such as GeRoMeSuite in (Quix et al., 2011) and LogMap in (Jiménez-Ruiz et al., 2015).

The choice of the anchoring matcher is a compromise between the efficiency and the effectiveness. Indeed, using a simple matcher is faster than using a complex one but less effective. In (Sabou et al., 2008), the authors have evaluated the impact of the anchoring matcher on the BK-based matching result. They implemented two matchers. The first one used only a strict string equivalence, while the second normalizes the concept labels, deals with compound names (e.g., different order of the label terms) and exploits WordNet to extract semantic relations between terms. The experimental results showed that the second matcher generated more correct mappings than the first one. However, the anchoring matcher should ensure a high precision, otherwise the BK-based matching may return more incorrect mappings than correct ones. The experiments conducted by (Safar et al., 2007) compared the label inclusion to strict string equivalence matchers. The use of the label inclusion matcher generated many incorrect mappings comparing to the strict string equivalence matcher.

As we have seen in Section 3.2, several BK selection methods considered the number of anchors between a given knowledge resource and the ontologies to align as a selection criteria of this knowledge resource. Hence, the anchoring is performed for selecting the BK and to exploit it.

Some works have reused existing alignments that include mappings between the ontologies to align and the selected BK. Hence, the anchoring step was ignored (Groß et al., 2011, Annane et al., 2016a) in BK exploitation.

3.3.2 Derivation

In this step, the selected BK is used as a mediator to derive mappings between the ontologies to be aligned. In the following, we will review the various derivation strategies according to the number of knowledge resources used as background knowledge, as well as the exploration or not of their structure during the derivation process.

3.3.2.1 Using one knowledge resource without structure exploration

Once the anchoring step is performed, the selected knowledge resources –the BK– is no more used. The mapping derivation consists in composing the mappings resulted from anchoring without exploiting the relations linking the BK entities. We present an example in Figure 3.2 (a) to illustrate this derivation strategy. As we can see, the anchoring produced four mappings $m_1 =< C_{S1}, C_{BK1}, \equiv >$, $m_2 =< C_{T1}, C_{BK1}, \equiv >$, $m_3 =< C_{S2}, C_{BK2}, \equiv >$, $m_4 =< C_{T2}, C_{BK3}, \subseteq >$. Only m_1 and m_2 are composed

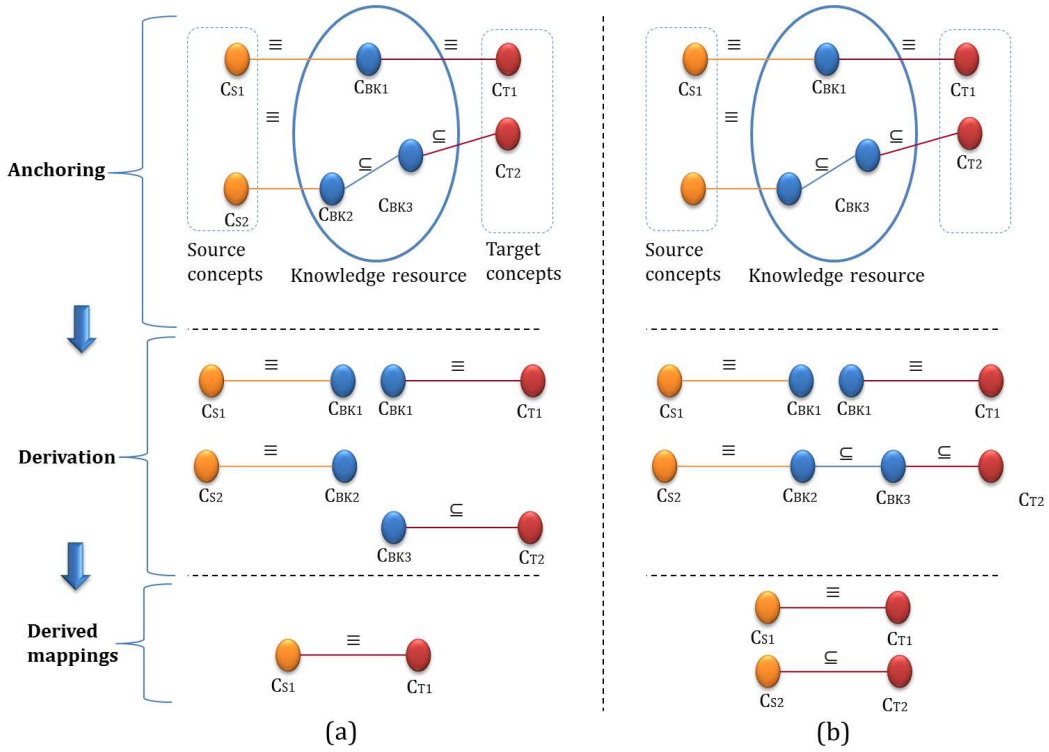

Figure 3.2: Mapping derivation using one knowledge resource and (a) with/ (b) without structure exploration.

because they have a common anchor C_{BK1}. This composition derives one mapping between the source and target ontologies ($m =< C_{S1}, C_{T1}, \equiv >$).

This derivation strategy has been implemented in (Mascardi et al., 2010). The authors have exploited an upper level ontology as background knowledge. They derived equivalence mappings, where the score was the multiplication of the composed mapping scores. The union of all the derived mappings constituted the final alignment. The authors have evaluated their method by matching 17 small ontologies (<200 concepts) downloaded from the web. According to the experiment results, the use of upper ontologies as background knowledge ensures an improvement of the direct matching F-measure at the price of a long alignment process (several hours) because of their large size.

3.3.2.2 Using one knowledge resource with structure exploration

In this derivation strategy, the internal relations between the knowledge resource entities (e.g., subClass, disjoint) are explored during the derivation process. On the opposite of the previous approach, the derivation requires to reuse the knowledge resources after the anchoring step. This process is illustrated in in Figure 3.2 (b),

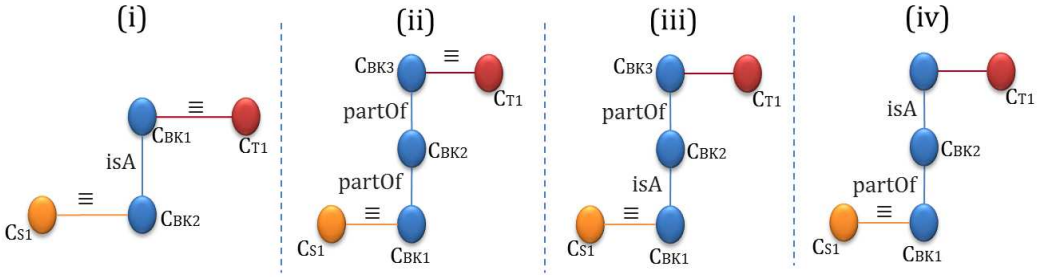

Figure 3.3: Mapping derivation strategies (Aleksovski et al., 2006b)

the mapping derivation step takes into account the subClass relation between the concepts C_{BK3} and C_{BK2} which allows to derive the mapping $< C_{S2}, C_{T2}, \subseteq >$. Note that in Figure 3.2 (a) this mapping has not been derived because the derivation is performed without exploring the BK structure.

Methods exploiting the BK structure varies according to the internal BK relations considered within the derivation process, and the technique used to compose relations of the anchoring mappings with those of the BK.

In (Aleksovski et al., 2006a), to match two biomedical vocabularies (flat lists), the authors exploited DICE –a biomedical ontology– as background knowledge. They have explored the structure of the BK ontology through an internal relation, named *kinds-Of*, to derive mappings of various types: subsumes, subsumed by, and equivalence mappings. The experiments showed good results with 76% correct mappings and 24% wrong ones vs. 46% and 54% when using the direct matching approach. Advanced experiments have been conducted in (Aleksovski et al., 2006b) to confirm the effectiveness of the proposed method. The two anatomy parts of the biomedical ontologies CRISP[1] and MeSH[2] have been matched exploiting the FMA[3] ontology as a BK. The derivation considered *isA* and *partOf* relations of the BK ontology –FMA ontology. Both of these relations are specializations of the *broaderThan* relation, hence they had the same representation (i.e., C_1 isA $C_2 \rightarrow C_1 \subseteq C_2$ and C_1 partOf $C_2 \rightarrow C_1 \subseteq C_2$). The authors evaluated several strategies depending on the composition of the *isA* and *partOf* relations: (i) without composition: only one internal relation –isA or partOf– should link two anchors, (ii) several relations may link two anchors but they have to be of the same type, (iii) both relations are composed with each other in any order , and finally (iv) composing the two relations such that there is no isA relations before the partOf relations. In Figure 3.3, the different strategies are illustrated, and derive the same mapping $< C_{S1}, C_{T1}, \subseteq >$. The best results were obtained using the last strategy.

In addition to the ontologies, WordNet (Miller, 1995) is widely used as an external knowledge resource. It is an English lexical resource that groups terms (nouns,

[1]Computer Retrieval of Information on Scientific Projects
[2]Medical Subject Headings
[3]Foundational Model of Anatomy

verbs, adjectives and adverbs) in sets of synonym called *Synsets*. The *Synsets* are related by relationships such as *isA, kindOf*. There are two strategies to exploit WordNet in ontology matching. The first one consists in extending concept labels with synonyms (Lin and Sandkuhl, 2008). The second one, on which we focus in this thesis, considers WordNet as a hierarchy of concepts and exploits the relations between these concepts to derive mappings, which is the case of the work presented in (Safar et al., 2007, Reynaud and Safar, 2007). To solve the ambiguity problem in WordNet (i.e., one label may have several senses), a human expert should select from WordNet one or several concepts –called roots– that cover the domain of the ontologies to align. Then, only descendants of the selected roots are considered within the derivation process. To derive mappings, the authors used two techniques: (i) A semantic technique that consists in finding the first target concepts in paths leading to the roots; (ii) A structural technique based on the *Wu and Palmer* (Wu and Palmer, 1994) which is a WordNet node similarity measure (Pedersen et al., 2004). The mappings generated with these techniques had isA and isClose relations, respectively.

Another mapping derivation approach has been investigated in (Giunchiglia et al., 2004). The ontology matching problem is transformed into a propositional satisfiability one, and a SAT Solver is used to derive mappings between the source and target concepts. *S-Match* is an ontology matching algorithm based on this approach with WordNet as background knowledge. S-Match includes two phases. In the first one, concept labels of the ontologies to be aligned are tokenized, lemmatized, and anchored to WordNet. Then, each concept is represented by a propositional expression using its labels and specific rules that translate prepositions, conjunctions, etc. into logical connectives. Additional axioms are generated using WordNet. For instance, if A and B are connected by the synonymy relation in WordNet, the axiom $A \leftrightarrow B$ is generated, where A and B are terms of the source and target ontologies, respectively. In the second phase, the SAT Solver used the generated axioms to verify the accuracy of a given mapping. For example the mapping $< C_S, C_T, \subseteq >$ is considered as correct if the implication $C_S \rightarrow C_T$ is validated by the SAT solver. S-Match does not need a complete lexical overlap between the ontologies to be aligned and the BK used, which is an advantage. S-Match generates mappings of different relation types: disjoint, subsumedBy and equivalence. According to the presented experiments, and comparing to the state-of-the-art systems, S-Match is effective but not efficient. Indeed, the mapping derivation using the SAT solver is time consuming.

S-Match algorithm has been adapted to match biomedical ontologies in (Shamdasani et al., 2009) where WordNet is replaced by a specialized domain knowledge resource: the Unified Medical Language System Meta-thesaurus (UMLS). UMLS is a lexical resource aggregating multiple biomedical ontologies and terminologies (Bodenreider, 2004). Since UMLS does not explicitly state antonymy between concepts as WordNet does, disjoint mappings have not been derived in this version.

The approaches using the resource structure within the derivation process gener-

ate mappings of various relations, while the approaches that do not use the resource structure generate only equivalence mappings.

3.3.2.3 Using several knowledge resources without structure exploration

This derivation strategy reuses the technique explained in Section 3.3.2.1 for each BK resource. It is implemented by several ontology matching systems such as LogMap-Bio (Jiménez-Ruiz et al., 2015) and AML (Faria et al., 2013b), which exploit a set of ontologies as BK to generate equivalence mappings. The GOMMA system (Groß et al., 2011, Groß et al., 2012) reused existing mappings as BK. Indeed, with this derivation strategy, no need to process the selected BK within the derivation process.

All the ontology matching systems cited previously compose only anchoring mappings related to the same BK ontology, they do not derive mappings across several intermediate ontologies (see Figure 3.4 (a)). However, in (Annane et al., 2016a), in addition to the anchoring mappings, the authors exploited alignments produced by matching BK ontologies between each other. For instance, this permits to generate the mapping between C_{S2} and C_{T2} in Figure 3.4 (b). The exploited mappings in (Annane et al., 2016a) were extracted from the repository of biomedical ontologies NCBO BioPortal (Noy et al., 2009), and they were of equivalence type but of various provenances. These were produced either manually by human experts or automatically by a simple syntactic matcher called *LOOM* (Ghazvinian et al., 2009b). The authors proposed to combine these mappings in one global graph, called *Global Mapping Graph*, where nodes were concepts of different ontologies and edges were various mappings linking these concepts. The edges were tagged with the mapping provenance and no score value was used. First, only the source ontology is anchored to the *Global Mapping Graph* to select a customized fragment called *Specific Mapping Graph*. Then, the target ontology is anchored to the selected fragment to allow deriving equivalence mappings between the source and target concepts.

3.3.2.4 Using several knowledge resources with structure exploration

This derivation strategy reuses the technique explained in Section 3.3.2.2 for each BK resource.

In (Quix et al., 2011), the authors used a set of ontologies as background knowledge. Each BK ontology has been exploited separately to derive mappings. The structure of the BK ontologies were explored via the isA relation.

In the previous work (Quix et al., 2011), the derivation strategy assumes that the mapping between a pair of source and target concepts should be covered by one BK ontology. This assumption has been also experimented in (Sabou et al., 2008) and compared to another one which assumes that the relation may be distributed over several ontologies – derivation across several BK ontologies. The result comparison of the two assumptions showed that the derivation across several BK ontologies is more effective, i.e. it finds more correct mappings.

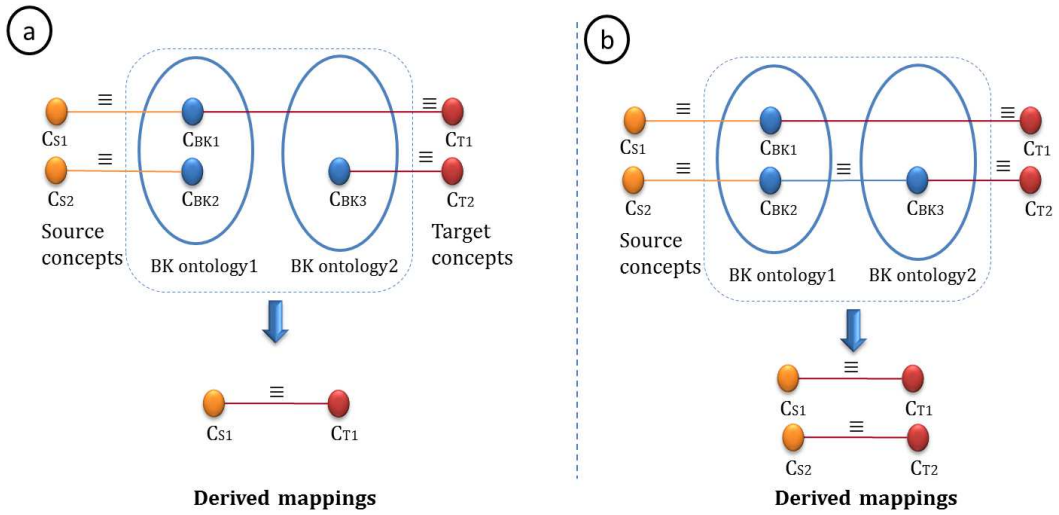

Figure 3.4: Derivation cross several BK ontologies and (a) with/ (b) without structure exploration.

The work described in (Sabou et al., 2008) has been extended in (Locoro et al., 2014) to a generic framework, called *Scarlet2.0*. In this framework, the derivation step was controlled by various parameters such as: the *exploration type* that specifies the relation to be used to explore the BK ontology structure; and the *maximum global path* to limit the maximum number of BK ontologies that can link a source concept to a target one. The relation composition to derive mappings was performed using the algebra of relations defined in (Euzenat, 2008). The experimental results confirmed that deriving mappings across several BK ontologies increases the recall.

Ontologies or mappings are structured knowledge resources whereas text is an unstructured one. Textual resources require more treatments to be exploited as BK. They should be structured beforehand, which is not an easy task. Only few works have dealt with such type of knowledge resources. For instance, (Kingkaew, 2012) has attempted to use the text content of web pages. The web pages, which were automatically selected, are given as input to a natural language processing component that extracted concepts and linked them by similarities in a customized graph, called *Similarity Graph* (see Section 3.2 for more details about the automatic selection of the web pages). Once this graph is built, source and target ontologies are anchored to it using a lexical matcher. Then, mappings are derived between source and target concepts using paths in the Similarity Graph. According to the experimental results, this method provides good results but building the similarity graph with the natural-language processing component is complex and time-consuming.

For which entities the selected BK should be exploited? In (Reynaud and Safar, 2007), BK-based matching concerned only source concepts that have not been mapped with the direct matching approach, while in (Groß et al., 2011), the method starts by matching the ontologies indirectly exploiting the BK then, source and target concepts that have not been mapped were extracted and matched directly. In other works, whole ontology entities are matched directly and indirectly, then the results are aggregated (Quix et al., 2011, Locoro et al., 2014). According to the empirical results of these works, exploiting BK improves the direct-alignment quality (i.e., Recall and F-measure), however, no evaluation has been made to highlight the best BK exploitation strategy.

3.3.3 Aggregation and Selection

Aggregation. Depending on the mapping cardinality generated by the anchoring matcher (i.e., one to one, one to many, or many to many), and the number of knowledge resources exploited in the matching process, for the same pair of source and target concepts, we may derive various mapping relations (Sabou et al., 2008). The simplest aggregation strategy is the union of all the derived mappings (Quix et al., 2011, Groß et al., 2011), however, in some cases more sophisticated aggregation strategies are required. For instance, the union strategy is not appropriate to aggregate the two mappings $m_1 =< C_S, C_T, \subseteq >$ and $m_2 =< C_S, C_T, \perp >$ derived using two knowledge resources KR_1 and KR_2, respectively. In (Locoro et al., 2014), the authors proposed to use other strategies such as algebraic operations (i.e., conjunction, disjunction) or popularity. The popularity aggregation keeps the most frequent mapping relation obtained between a given pair of source and target concepts.

Selection. In the first BK-based matching works, all the derived mappings were returned without a specific selection strategy (Aleksovski et al., 2006b, Sabou et al., 2008). However, the derived mappings are not always correct, even when using knowledge resource of high quality such as ontologies (Tordai et al., 2010). Hence, the need of an effective mapping selection strategy after the derivation step.

In (Groß et al., 2011), the selection of the final mappings is controlled by the minimum occurrence necessary for a given mapping to be selected in the final alignment.

Selecting the most relevant mappings among the candidate ones is a common task between direct and BK-based matching. Indeed, ontology matching systems, which implement a BK-based component, aggregate the direct matching candidate mappings, with the derived ones before applying selection strategies (Quix et al., 2011, Faria et al., 2013b). In the following, we will describe briefly some mapping selection strategies implemented in direct matching methods.

Threshold filter. A threshold filter is a simple filter that selects mappings having confidence values equal to or higher than a predefined threshold value. Indeed, the confidence value reflects the degree of confidence that two entities are similar, the

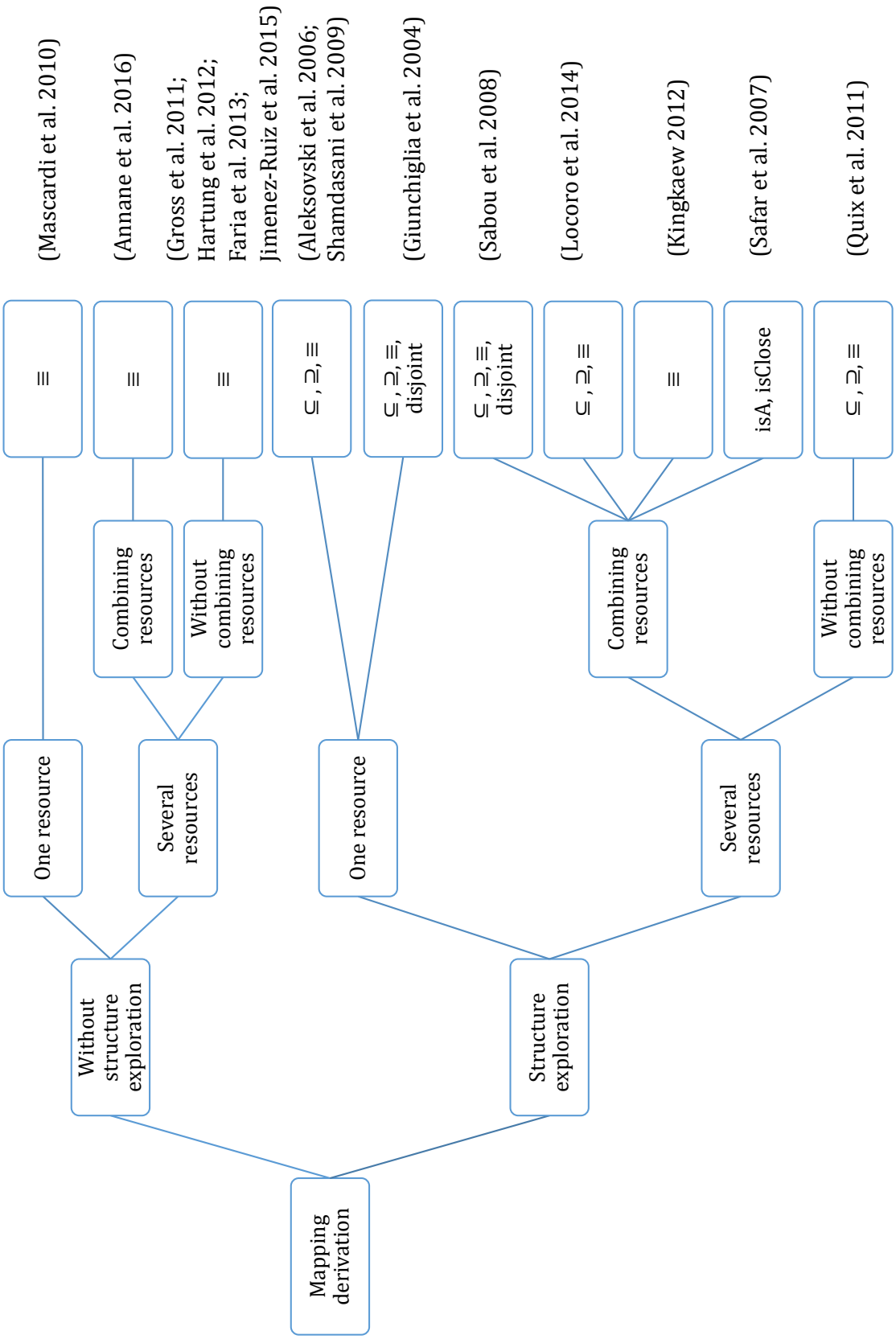

Figure 3.5: Classification of mapping derivation methods (Symbols $\supseteq, \subseteq, \equiv$ represent respectively *subsumes*, *subsumed by* and *equivalence* mappings.)

higher the confidence value is, the more likely two entities are matched. However, confidence values given to the same pair of entities are different from one matcher to another, which makes the task of identifying the best threshold value for a given matcher and a matching task challenging.

Semantic selection. The semantic selection deals with logical inconsistencies. Input ontologies and candidate mappings are interpreted as sets of axioms in description logic, and logical reasoners are applied to detect inconsistency (Jiménez-Ruiz et al., 2013). Thus, inconsistent candidate mappings are detected and eliminated to generate the final alignment. After having applied such a selection method, merging the input ontologies using the resulted alignment should be consistent.

Supervised machine learning.The mapping selection problem may be transformed to a classification problem. Indeed, deciding to keep or not a given candidate mapping in the final alignment may be seen as classifying a candidate mapping as *correct* or *incorrect*. The general workflow of supervised machine learning is generic, but it requires the definition of two key parameters: the attributes that describe the training data and test data objects, and how to obtain or generate the training data (classified objects). Syntactic, structural and lexical similarity measures are used as attributes to describe a candidate mapping. However, the generation of training data varies from one work to another. (Spohr et al., 2011a) proposed to use manually produced mappings as training data. This technique was criticized because it is based on the cognitive abilities in terms of memory and decision-making of the user that should manually create the mappings. In addition, it is fastidious, and may not provide enough data to learn an effective classifier (Dragisic et al., 2016). Other tools such as APFEL (Ehrig et al., 2005) generates mappings automatically, and asks a user to validate them. Both correct and incorrect mappings are used as training data. Despite the drawbacks of generating the training data by users, the advantage is that user preferences are captured from his mappings. Another solution was adopted in (Rong et al., 2012) considers mappings that are generated automatically as training data. This technique generates larger training data than the previous one. However, it does not take into account the user preferences. Moreover, it may consider incorrect mappings as correct, which affects the learned classifier.

3.4 Evaluation and comparison

The aim of this section is to provide a comparative review of the BK-based matching systems by analyzing their performance in order to evaluate the benefit of using BK. For this purpose, and for the sake of a fair comparison, we extract their results (i.e., Precision, Recall and F-measure values) from the Ontology Alignment Evaluation Initiative (OAEI[4]) campaigns. More precisely, we consider here the two tracks of OAEI in which the participating systems have used BK to enhance the matching results: the Anatomy (see Section 3.4.2) and LargeBio (see Section 3.4.3) tracks.

[4]http://oaei.ontologymatching.org/

The LargeBio track has been initiated in 2012, consequently our study concerns the 2012-2016 OAEI campaigns.

In the following, we present the ontology matching systems using BK that participated in the studied campaigns. Then, we describe the datasets on which the evaluation was done and we compare the obtained results.

3.4.1 Ontology matching systems using BK

GOMMA-BK(Generic Ontology Matching and mapping Management). It is the first system that has implemented BK-based approach in 2012 by using mappings composition (Groß et al., 2011)(see Section 3.3).

AML-BK. A version of AgreementMakerLight ontology matching system (Faria et al., 2013b). AML-BK used the Uber Anatomy Ontology (UBERON) ontology as BK in 2013. Since 2014, it uses two biomedical ontologies as a knowledge resource pool that are : UBERON, Human Disease Ontology (DOID). For each ontology matching task, AML-BK selects automatically using the mapping gain measure the ontologies to be exploited as BK from its predefined knowledge resource pool. In addition, AML takes as input the Lexicon file of the Medical Subject Headings (MeSH) ontology.

LogMap-BK/LogMapBio. They are two versions of the LogMap ontology matching system that used BK. LogMap-BK used UMLS Lexicon While LogMapBio includes an extension for selecting automatically a set of biomedical ontologies as BK from NCBO BioPortal (Chen et al., 2014)(See Section 3.2).

Note that, in 2016, AML did not use anymore the suffix BK in its name even if the system actually used BK. For better readability, we have explicitly post fixed its name with BK (same thing for LogMap in 2012, 2015, 2016).

3.4.2 Anatomy track

The Anatomy track consists in finding an alignment of $1,516$ mappings between the Adult Mouse Anatomy (2,744 classes) and a subset of the National Cancer Institute (NCI) Thesaurus ($3,304$ classes) describing human anatomy (Dragisic et al., 2017).

Figure 3.6 shows the results of the systems using BK for the Anatomy track from 2012 to 2016. We added the result of the best system that do not use BK (tagged with BSW) to compare with their results.

In 2012, GOMMA reused mappings to three external resources (UMLS, UBERON and FMA), which increases its F-measure to 0.923, keeping an acceptable execution time, while the best system that did not use specialized knowledge resources, YAM++ (Ngo and Bellahsene, 2016), had only 0.898.

In 2013, AML-BK implemented the mapping composition technique, which allowed it to be the top ranked system with an F-measure of 0.942.

We observe that even if AML-BK used only UBERON ontology as BK, it had a higher F-measure value than GOMMA-BK that used two biomedical ontologies in

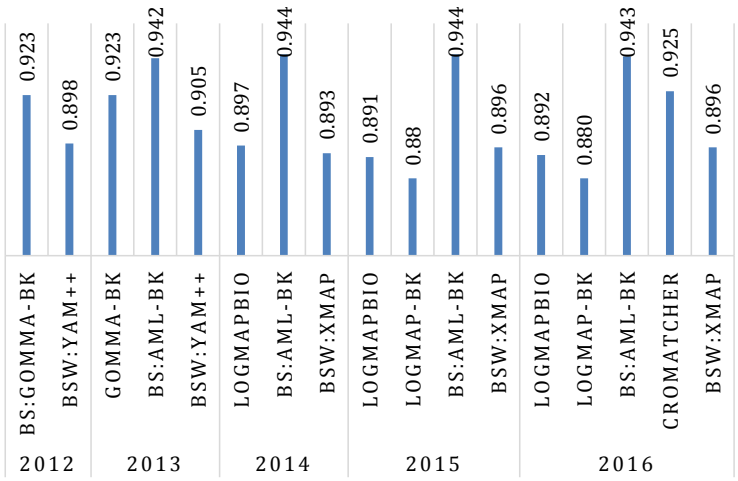

Figure 3.6: Anatomy track matching quality results (BS: Best system using BK; BSW: Best System Without BK)

addition to UBERON. This shows that the final results do not depend only on the BK used, but on the whole matching process implemented in each system.

From 2014 to 2016, LogMapBio participated in the Anatomy track, and obtained almost the same F-measure value of 0.891. It is the lowest F-measure of systems using BK, and lower than the F-measure value of the best system without BK (i.e., XMAP), which had an F-measure of 0.896.

In 2016, for the first time, the CroMatcher system used the UBERON ontology as background knowledge, which improved its F-measure value from 0.861 in 2015 to 0.925 in 2016.

Figure 3.7 shows the evolution of the Anatomy track results over the last four years (from 2012 to 2016). As we can see, from 2012 to 2016 the best F-measure value is obtained systematically by the BK-based systems. In particular, by using a specialized domain knowledge resources (i.e., biomedical ontologies). The F-measure values have stabilized from 2013 to 2016, AML-BK system dominates the task with almost the same F-measure value every year. The same thing for the results of the systems that do not use BK. The best F-measure value is obtained by the XMAP system since 2014.

3.4.3 Large Biomedical track

The Large Biomedical (LargeBio) OAEI track[5] aims at finding alignments between several large and semantically rich biomedical ontologies: the Foundational Model of Anatomy (FMA) (Rosse and Mejino, 2003), National Cancer Institute Thesaurus (NCI) (Sioutos et al., 2007) and SNOMED Clinical Terms (SNOMED-CT) (Donnelly, 2006), which contain 78,989, 66,724 and 306,591 concepts, respectively. In

[5]http://www.cs.ox.ac.uk/isg/projects/SEALS/oaei/

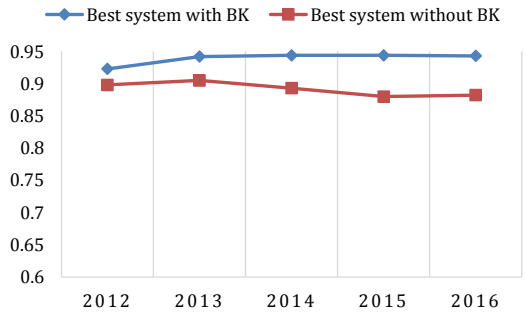

Figure 3.7: Evolution of the Anatomy track results.

Table 3.1 , we present the six matching tasks of LargeBio corresponding to the different sizes of input ontologies (small fragments/whole ontology of FMA and NCI and small/large fragments of SNOMED-CT). The last column shows the number of mappings in the reference alignment. The Unified Medical Language System (UMLS) (Bodenreider, 2004) has been used as the basis to produce the reference alignments (Cheatham et al., 2015).

Table 3.1: Matching tasks of the OAEI LargeBio track.

Task #	Task name	#Source	#Target	#Mappings
Task 1	FMA-NCI small fragments	3,696	6,488	2,686
Task 2	NCI-FMA Whole ontologies	66,724	78,989	2,686
Task 3	FMA-SNOMED small fragments	10,157	13,412	6,026
Task 4	FMA whole with SNOMED large fragment	78,989	122,464	6,026
Task 5	NCI-SNOMED small fragments	23,958	51,128	17,210
Task 6	NCI whole with SNOMED large fragment	66,724	122,464	17,210

Figure 3.8 and Figure 3.9 depict the LargBio sub-tasks results obtained by BK-based ontology matching systems from 2012 to 2016.

2012. Two systems using BK have participated to this track: GOMMA-BK and LogMap-BK. GOMMA-BK obtained the best F-measure value in Task 1 and Task 2, while LogMap-BK was the top ranked system in Task 5. In the large fragment tasks (i.e., Task 2, 4, and 6), the systems that do not use BK (YAM++ and ServOMapL) were more effective.

2013. In addition to LogMap-BK and GOMMA-BK, AML participated with a BK-based version for the first time. AML-BK obtained the best F-measure value in Task 1 and 5. It had also, a very close F-measure to the best one in Task 3 that was obtained by LogMap-BK. For the large fragment tasks, the best results was obtained by the systems that do not use BK (i.e., YAM++ , LogMap and ServOMap).

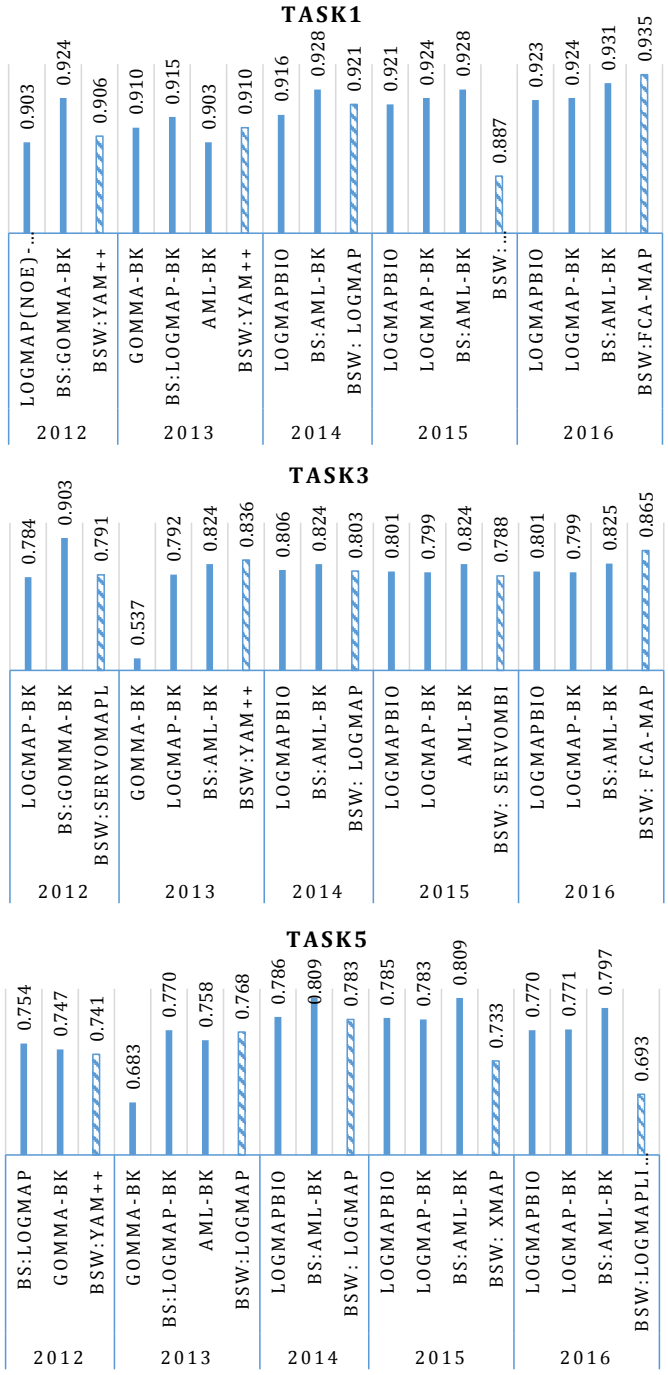

Figure 3.8: LargeBio track results (small fragments).

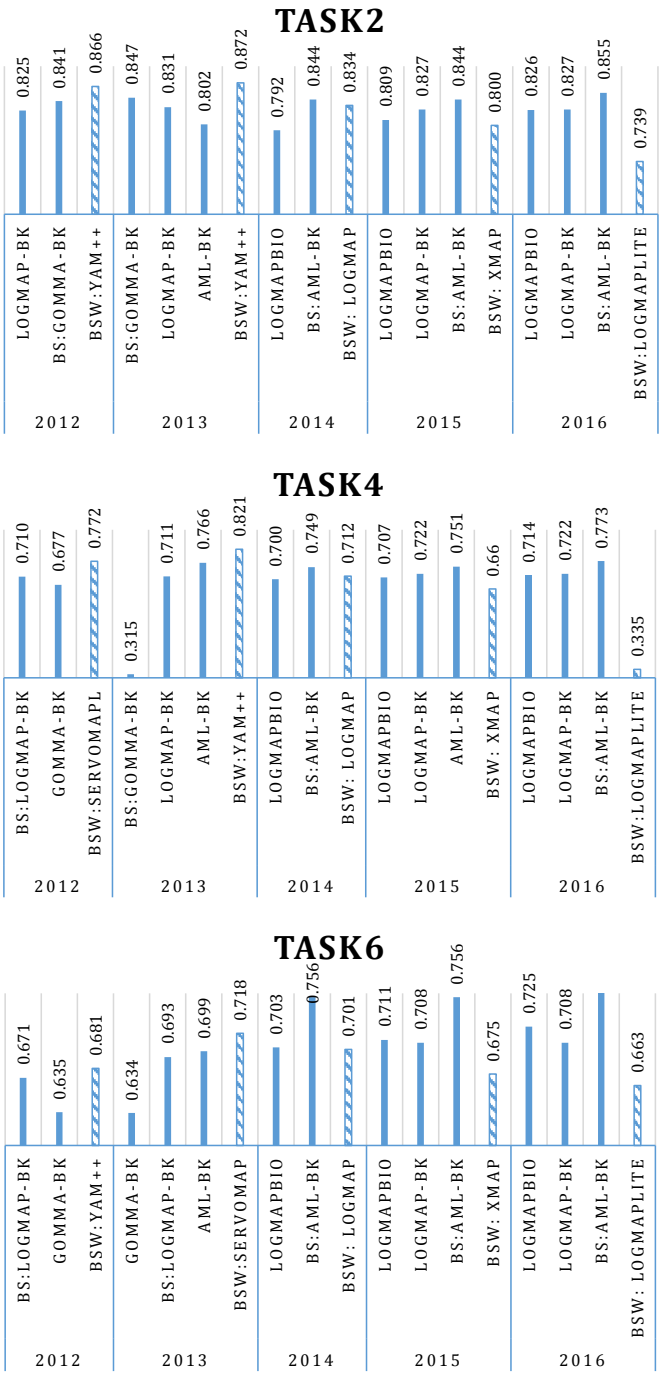

Figure 3.9: LargeBio track results (Large fragments).

2014. LogMap-BK has been replaced by LogMapBio that has not used the UMLS Lexicon as the previous years, instead, it selected dynamically a set of biomedical ontologies as BK from NCBO BioPortal. The systems that use BK have obtained the best results, in particular AML-BK which was the top ranked in all LargeBio tasks. We note also the absence of the three systems GOMMA-BK, YAM++ and ServOMap.

2015. AML-BK maintained its first position and results in the LargeBio tasks.

2016. The same BK-based systems from the previous year have participated, and obtained the best F-measure values in the six LargeBio tasks.

To observe the evolution of the LargeBio track performances in terms of F-measure over the last four years, we have represented in Figure 3.10 the best F-measures obtained by the systems using BK and the best F-measure values of the systems that did not use BK. As we can see in Figure 3.10, the exploitation of the BK for the small fragment tasks, especially Task 1 and 3, increases always slightly the F-measure values. The results of the same tasks obtained by the systems that do not use BK have been improved and converge to those obtained using specialized BK. For instance, in 2016 FCA-Map has obtained the best F-measure values in Task1, higher than the AML-BK and LogMapBio F-measure value (see Figure 3.8).

In 2012 and 2013, for large fragments tasks (i.e., Task 2, 4 and 6), we observe that the systems that do not use BK (YAM++ and ServOMap) have obtained the best results. From 2014 to 2016, YAM++ and ServOMap did not participate anymore and the systems with BK had the best results.

Task 5 represents the largest one in small fragment tasks in terms of concepts number (see Table 3.1). In 2012 and 2013, the results of the systems with BK are slightly better than those of the systems that did not use BK. However, from 2014 we observe a divergence. In particular, results of the systems that do not use BK are decreasing, and results of BK-based systems are stable.

According to this analysis and the results presented in Figure 3.10, we may derive that the use of BK enhances the quality of LargeBio task alignments. The trend of the last four years shows that the use of BK is more effective for large fragment tasks. However, for small fragments tasks, the effectiveness of BK exploitation seems to be limited since the systems without BK obtained the best or very close to the best F-measure values.

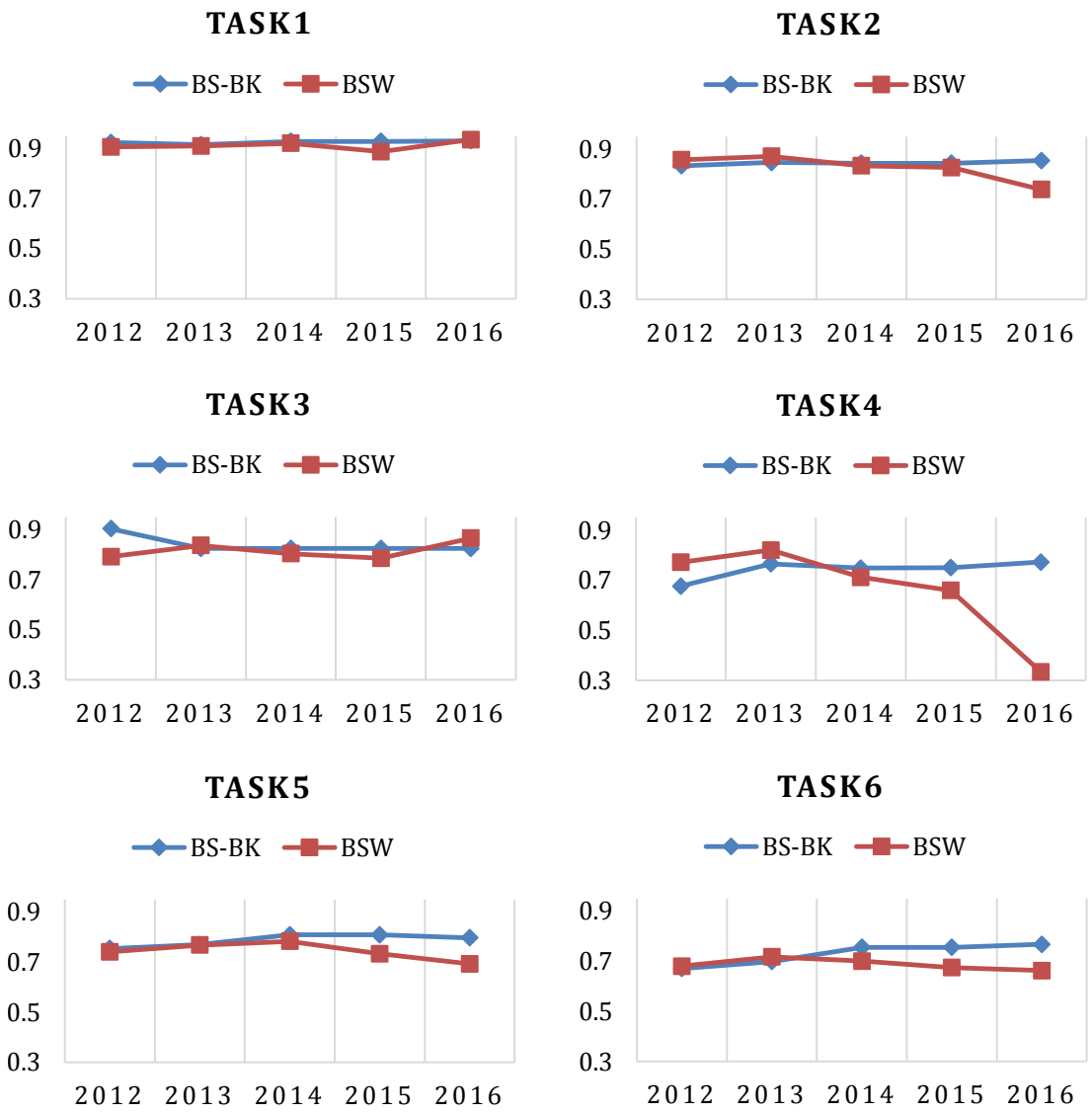

Figure 3.10: Evolution of the LargeBio track results.

3.4.4 Computation-time vs. F-measure improvement

The computation time is an important evaluation criteria of the matching strategy. The aim of this section is to attempt to evaluate the cost of using BK in terms of computation time. For this purpose, we compare the performance of the systems with and without BK. For a fair comparison, we do not compare the different systems to each other, we compare the different versions of the same system.

In Table 3.2, we summarize the computation time and matching quality of the systems that participated in Anatomy and LargeBio tracks with two versions from 2012 to 2016: GOMMA and GOMMA-BK in 2012 (Groß et al., 2012), AML and AML-BK in 2013 (Faria et al., 2013a), LogMap and LogMap-BK in 2013, and finally LogMap versus LogMapBio in 2016. Computation time is in seconds and the F-measure Gain is computed as follows:

F-measure Gain= (F-measure BK - F-measure)/F-measure)

As we can observe in Table 3.2, in almost all cases BK-based versions generated alignments with higher F-measure values. The exploitation of BK allows to discover more correct mappings, which enhances recall and F-measure.

The versions using BK take slightly more time to accomplish a given matching task than the versions that do not use BK.

In 2016, the gain is not significant for LogMapBio comparing to LogMap. This may be explained by the fact that LogMap used also UMLS Lexicon. The impact of using BK is negative in Task 1, 2 and 4. Indeed, for these tasks the use of BK leads to a low precision that decreases the F-measure. LogMapBio takes too much time to accomplish the LargeBio tasks with respect to the other systems because it does an automatic selection from an external web repository (NCBO BioPortal). LogMapBio selected 10 ontologies for each task from the NCBO BioPortal which contains more than 500 ontologies. It then downloaded the selected ontologies to exploit them. This process explains why LogMapBio spent much more time to complete the tasks when comparing to the LogMap system and AML that selects BK from only two preselected ontologies.

Table 3.2 shows that the exploitation of BK enhances the ontology matching quality mainly by increasing recall. The computation time in the BK-based versions varies from one system to another. Indeed, it depends on the size of the knowledge resource pool, and the methods implemented to select and exploit the BK.

Table 3.2: Performance analysis

Year	System	Parameter	Anatomy	Task1	Task2	Task3	Task4	Task5	Task6
		Time (s)	17	26	217	54	1994	197	1820
		Time BK (s)	15	26	231	148	1893	226	1940
		Precision	0.956	0.973	0.865	0.926	0.370	0.948	0.720
		Precision BK	0.917	0.961	0.818	0.958	0.602	0.946	0.669
2012	GOMMA	Recall	0.797	0.845	0.813	0.377	0.239	0.531	0.523
		Recall BK	0.928	0.903	0.856	0.914	0.858	0.617	0.605
		F-measure	0.870	0.904	0.839	0.536	0.291	0.68	0.606
		F-measure BK	0.923	0.931	0.837	0.935	0.708	0.747	0.635
		F-measure Gain	**6.09%**	**2.99%**	**-0.24%**	**74.44%**	**143.30%**	**9.85%**	**4.79%**
		Time (s)	15	16	202	60	542	291	570
		Time BK (s)	43	39	201	93	530	380	571
		Precision	0.954	0.947	0.880	0.943	0.937	0.895	0.926
		Precision BK	0.954	0.942	0.816	0.942	0.963	0.894	0.918
2013	AML	Recall	0.827	0.834	0.730	0.720	0.624	0.642	0.549
		Recall BK	0.929	0.867	0.787	0.731	0.648	0.658	0.564
		F-measure	0.886	0.887	0.798	0.816	0.758	0.747	0.689
		F-measure BK	0.942	0.903	0.802	0.824	0.766	0.758	0.699
		F-measure Gain	**6.32%**	**1.80%**	**0.50%**	**0.98%**	**1.06%**	**1.47%**	**1.45%**
		Time (s)		41	162	79	537	433	2485
		Time BK (s)		45	173	85	556	444	2391
		Precision		0.952	0.874	0.966	0.888	0.896	0.910
		Precision BK		0.949	0.872	0.963	0.874	0.894	0.904
2013	LogMap	Recall		0.851	0.795	0.656	0.588	0.672	0.689
		Recall BK		0.883	0.794	0.672	0.600	0.677	0.700
		F-measure		0.889	0.832	0.782	0.708	0.768	0.780
		F-measure BK		0.915	0.831	0.792	0.711	0.770	0.785
		F-measure Gain		**1.8%**	**-0.12%**	**1.28%**	**0.42%**	**0.026%**	**0.64%**

Year	System	Parameter	Anatomy	Task1	Task2	Task3	Task4	Task5	Task6
		Time (s)	24	10	80	60	433	177	699
		Time BK (s)	758	1712	1188	1180	2156	3757	4322
		Precision	0.918	0.949	0.854	0.948	0.839	0.922	0.87
		Precision BK	0.888	0.935	0.818	0.944	0.808	0.896	0.842
2016	LogMap/LogMapBio	Recall	0.846	0.901	0.802	0.69	0.634	0.663	0.596
		Recall BK	0.896	0.91	0.835	0.696	0.64	0.675	0.637
		F-measure	0.880	0.924	0.827	0.799	0.722	0.771	0.708
		F-measure BK	0.892	0.923	0.826	0.801	0.714	0.77	0.725
		F-measure Gain	**1.33%**	**-0.11%**	**-0.12%**	**0.25%**	**-1.11%**	**-0.13%**	**2.40%**

3.5 Discussion

BK-based ontology matching raises several questions. In the following, we will try to provide some answers to the questions raised in the introduction of this chapter according to our study.

In which cases is BK-based matching relevant and necessary?

The background knowledge resources play the role of a semantic bridge between the ontologies to be aligned. The exploitation of these resources allows to find new mappings missed by the direct matching methods. Exploiting BK is thus necessary in the presence of high lexical or structural heterogeneity between the ontologies to be aligned (e.g., equivalent concepts described with dissimilar labels). Furthermore, BK-based matching finds mappings with semantic relations such as less general, more general, etc. which are convenient for reasoning purposes.

What are the application domains in which such a matching approach can be used?

BK-based ontology matching approach is domain independent. It has been implemented to match ontologies of various domains (see Section 3.3) exploiting generic knowledge resources (e.g., WordNet, upper-level ontologies) or specialized ones (e.g., biomedical ontologies). However, experiments show that generic knowledge resources such as WordNet are prone to produce erroneous mappings in domains with specialized vocabularies, such as the biomedical domain (Faria et al., 2015). In this case, specialized knowledge resources seem to be more effective than the generic ones. Consequently, domains that promote BK-based matching are those rich in specialized and structured knowledge resources. For instance, this approach is widely adopted to match biomedical ontologies thanks to the profusion of knowledge resources in biomedicine (ontologies, terminologies and existing alignments).

Unstructured resources such as text may be an interesting alternative for domains that do not have this richness. However, they require the development of efficient and effective methods to structure them (i.e., extract entities and semantic relations linking them) into an effective knowledge resources.

What is the cost of the BK-based matching approach?

Exploiting external knowledge resources in ontology matching implies more computational time and memory resources (see Table 3.2). This cost depends on many factors, we can cite: (i) the number, the type (i.e., structured or not), the size, and the complexity of the knowledge resources in the selected BK and the BK resource pool in case of automatic BK selection; (ii) the location of these knowledge resources (e.g., the web or a local repository); and (iii) the methods used for BK selection and BK exploitation.

Another cost that should be highlighted, is the impact on the precision of the final alignments. Indeed, the BK-based matching approach may decrease the preci-

sion by generating additional incorrect mappings (see Section 3.4.4).

BK-based ontology matching an alternative or a complementary solution?

BK exploitation depends on the direct matching between the ontologies to align and the selected BK (anchoring). Hence, it cannot replace the direct matching methods. However, and as demonstrated by the experimental results of the reviewed works, it allows finding mappings that are missed by the direct matching due to the semantic heterogeneity i.e., it allows to complement the direct matching results. Indeed, as we have seen in Section 3.4, the BK-based systems have the best results. These systems uses a BK-based component as an extension in the biomedical tracks.

Moreover, BK-based matching depends on the quality and the availability of structured knowledge resources and their overlapping with the ontologies to be aligned; without such resources it cannot be effective.

3.6 Conclusion

The exploitation of external knowledge resources is one of the main ontology matching challenges (Shvaiko and Euzenat, 2013). In this chapter we have attempted to review the works related to this challenge according to the common workflow presented in the previous chapter. Moreover, we have elaborated two classifications to highlight the various BK selection and mapping derivation methods. Finally, we have compared and discussed the results of the BK-based ontology matching systems in OAEI from 2012 to 2016.

Chapter

4 BK Selection/Building

Contents

4.1 Introduction

As we have seen in the previous chapter, the automatic selection of ontologies as background knowledge has been proposed in several works (Sabou et al., 2008, Quix et al., 2011, Hartung et al., 2012, Faria et al., 2014). However, all the proposed methods return complete ontologies as background knowledge resources to be used in the matching process.

Our hypothesis is that within each effective BK ontology, especially large ones, only fragments are actually effective. Hence, the issue is that of the selection of these fragments from each BK ontology and their combination to build an efficient and effective BK.

In our BK selection approach, we tackle this issue by selecting only the concepts that are related to the matching task from the preselected ontologies. We then combine these selected concepts to build the BK: a novel knowledge resource (Annane et al., 2016a, Annane et al., 2018).

As we will experimentally demonstrate, following our approach, the built BK has a very reduced size comparing to that of the preselected ontologies, which improves the efficiency of the BK selection process. Furthermore, the built BK interconnects concepts from different preselected ontologies via mappings, thereby allowing deriving mappings across several intermediate ontologies.

The remainder of this chapter is organized as follows. Section 4.2 describes our BK selection approach and Section 4.3 demonstrates its efficiency. Then, Sections 4.4 and 4.5 present the experiment materials and the implementation techniques used for evaluation. Section 4.6 presents and discusses the obtained results. Finally, Section 4.7 concludes this chapter.

4.2 Description of our BK selection approach

Our BK selection approach includes four steps illustrated in Figure 4.1. In the following, we will describe these steps.

4.2.1 Ontology preselection

Today, a simple Google Search for "filetype:owl" returns around 34K results. Fittingly, these ontologies are often organized per domain or community in ontology libraries (Ying and Dieter, 2001, d'Aquin and Noy, 2012) such as the NCBO BioPortal, the AgroPortal (Jonquet et al., 2017) or the Marine Metadata Interoperability repository (Rueda et al., 2009). Ontology preselection consists in determining which ontologies to consider for the BK selection process among all the ontologies that exist. It aims at reducing the search space for the BK selection process by eliminating at the outset ontologies that would not be effective to identify new mappings (e.g., ontologies that are not of the same domain as the ontologies to align). The preselected ontologies may be an ontology repository (Chen et al., 2014), a specific set of

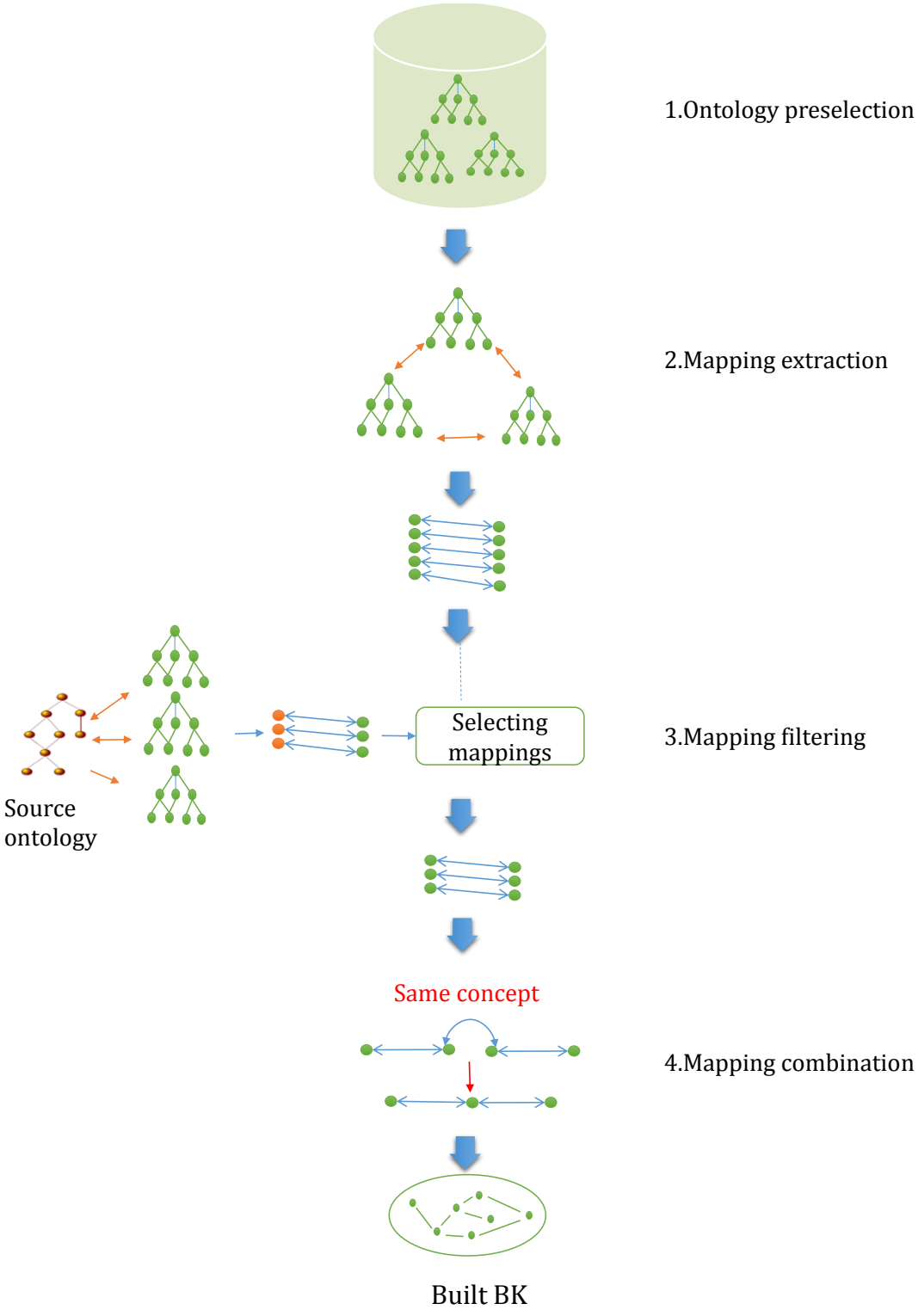

Figure 4.1: Overview of the BK selection process.

ontologies (Faria et al., 2014,Faria et al., 2016,Hartung et al., 2012) or all ontologies indexed by a given semantic web search engine (Sabou et al., 2008, Locoro et al., 2014).

In the related works, ontology preselection has not been formalized as a step of the BK-based ontology matching workflow, except in (Locoro et al., 2014) where ontology preselection was called *ontology arrangement*.

In our approach, and, at the best of our knowledge, in all related works, ontology preselection is performed manually (Sabou et al., 2008,Faria et al., 2014,Faria et al., 2016, Hartung et al., 2012,Locoro et al., 2014).

4.2.2 Mapping extraction

The experiments reported in (Ivanova and Lambrix, 2013,Sabou et al., 2008,Locoro et al., 2014) showed that combining several BK ontologies generates more correct mappings. Figure 4.2, sampled from our evaluation, illustrates this benefit. Each concept is represented with the term *ontology#ConceptIdentifier* and is interconnected with mappings. As we can see, the source and target concepts are linked via at least two intermediate concepts that belong to two different BK ontologies. Such correct mapping would not have been identified if we had used each intermediate ontology separately from the others (one intermediate concept at a time). Therefore, in this step, we extract all possible mappings between the preselected ontologies to be able to generate mappings across several intermediate concepts that belong to different BK ontologies.

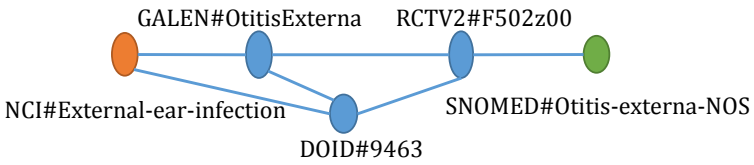

Figure 4.2: Example of a correct mapping between NCI and SNOMED derived across intermediate concepts from different BK ontologies.

Let $S = \{O_1, O_2, ..., O_n\}$ be the set of preselected ontologies. In this step, each ontology O_i in S is matched to the other preselected ontologies that have a higher index (i.e., O_{i+1}, O_{i+2},....,O_n). The matching of each couple of ontologies (O_i, O_j) provides an alignment that is a set of s mappings $A_{ij} = \{m_1, m_2, ..., m_s\}$. For n preselected ontologies, the result is the union of $\sum_{i=1}^{n-1}(n-i)$ alignments. More specifically, the result is the union of all mappings that compose the different alignments A_{ij}: $M = \bigcup_{i=1}^{n-1} \bigcup_{j=i+1}^{n} A_{ij}$.

The easiest way to extract these mappings is to use an automatic matcher. Several state-of-the-art matchers, such as YAM++ (Ngo and Bellahsene, 2016), LogMap (Jiménez-Ruiz et al., 2015), AML (Faria et al., 2013b), etc., are readily

available. As shown in the previous OAEI campaigns, these systems provide high-quality alignments (i.e., alignments with high F-measure score).

Furthermore, if available, mappings between the preselected ontologies that are manually created or human-curated should be added to the automatically extracted ones. For instance, in the biomedical domain, cross-references between OBO Foundry ontologies (Smith et al., 2007) may be considered as manual mappings.

Note that the mapping extraction task may be ignored if the preselected ontologies are not to be combined. In addition, this step is performed only once for a given set of preselected ontologies.

4.2.3 Mapping filtering

The preselected ontology concepts likely to generate new mappings should be related directly or indirectly to the source ontology. Conversely, those concepts not related to the source ontology will not help generate new mappings. Hence, it seems more efficient to eliminate the latter at the outset.

We start by matching the source ontology O_s to the preselected ontologies in S. In order to improve efficiency, the smallest of the ontologies to align is chosen as the source ontology. Indeed, the source ontology will be matched to all the preselected ontologies, while the target ontology will be matched only to the built BK (see the next section).

The mappings obtained by matching the source ontology to the preselected ontologies initialize the set of filtered mappings, noted FM. Recursively, we enrich FM by selecting all the mappings in M related to the target concepts of mappings already present in FM, and so on, until no new mapping is found in M. More precisely, until all mappings related to the source ontology in M are in FM. In each step, FM is enriched as follows:
$FM = FM \cup \{m_i / m_i \in M \text{ and } C_s(m_i) = C_t(m_j) \text{ and } m_j \in FM\}$

where $C_s(m_i)$ is a function that returns the source concept of the mapping m_i and $C_t(m_j)$ is a function that returns the target concept of the mapping m_j.

4.2.4 Mapping combination

Mappings filtered in the previous step are then combined in one unique graph where nodes are concepts and edges are mappings that link these concepts. This combination insures that each concept appears only once (i.e., mappings that share a concept are merged). Figure 4.3 shows an example of mapping combination. m_1 and m_2 are two mappings that have a common concept e_2. The combination keeps only one occurrence of the concept e_2. Note that, thanks to this combination, concepts that are not directly connected (e_1 and e_3 in Figure 4.3) may be indirectly connected through common concepts.

In the resulted graph, each selected concept (node) is described with four attributes:

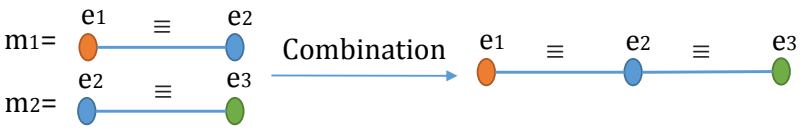

Figure 4.3: Mapping combination example.

1. the URI of the concept
2. the URI of the ontology to which the concept belongs
3. the preferred label of the concept
4. the concept synonyms

The mappings (or edges) between concepts are described with three attributes:

1. the source from which each mapping has been extracted. It may be the name of a resource such as UMLS, or the ontology matching tool name when the mapping was generated automatically.
2. the mapping score.
3. the type attribute that indicates whether the mapping was generated manually or automatically.

The generated graph is the *built BK* that will be used in the BK exploitation step.

4.3 Efficiency gain with the built BK

Building a new resource (i.e., the built BK) from the preselected ontologies is more efficient than returning complete ontologies as background knowledge. In this section, we estimate the computation time of our BK selection approach and that of the traditional approach, then we compare them to demonstrate the efficiency of our approach.

The traditional approach refers to the BK selection methods that match the source and target ontologies to all the preselected ontologies, and then they use the generated alignments to select the ontologies to be exploited as background knowledge (Hartung et al., 2012, Faria et al., 2014, Locoro et al., 2014).

Anchoring is the step that follows BK selection, its computation time depends on the selected BK: a set of ontologies or the built BK. Therefore, we include the anchoring computation time in our comparison. However, we do not include the mapping extraction computation time because it is performed once between the preselected ontologies independently of the matching tasks. Moreover, when comparing our approach to those that use each BK ontology separately (derivation across only one intermediate concept) (Hartung et al., 2012, Faria et al., 2014), the mapping

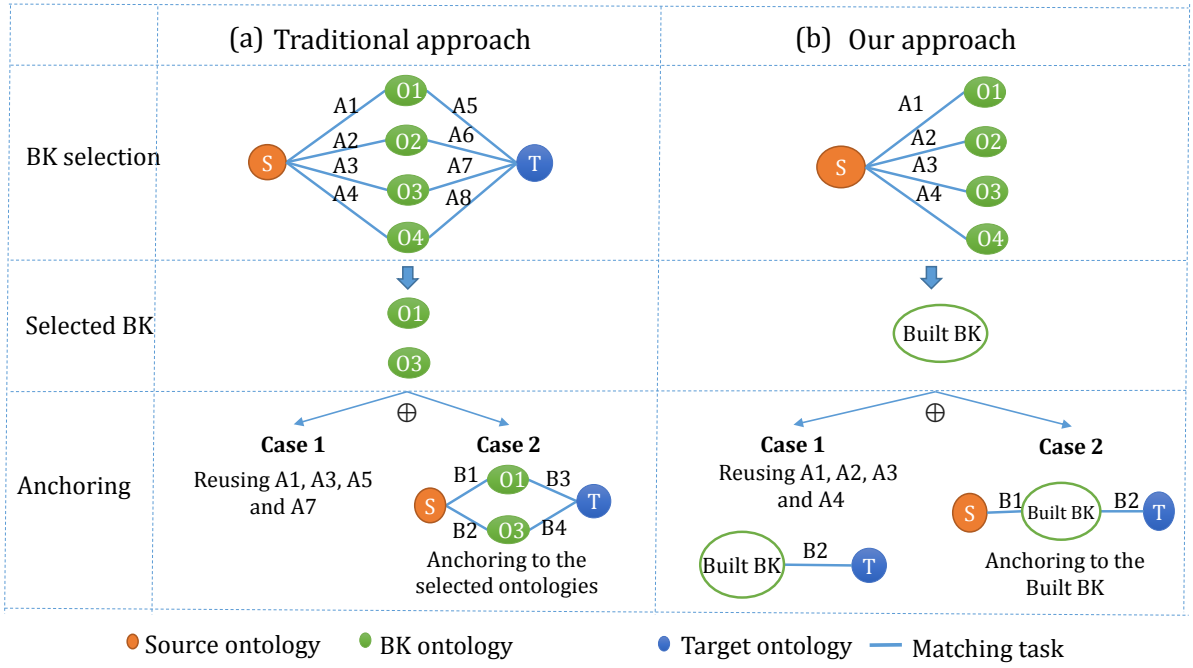

Figure 4.4: BK Selection and anchoring: Traditional approach vs. our approach.

extraction time has a zero value. Indeed, these works do not match BK ontologies between each other.

Let $KR = \{O_1, O_2, ..., O_n\}$ be the set of preselected ontologies, O_S the source ontology and O_T the target ontology, $t(M, O_1, O_2)$ the function that returns the time required by the matcher M to align the ontologies O_1 and O_2. When using the traditional approach, the selected BK is a set of k ontologies $SR = \{SO_1, ..., SO_k\}$, with $SR \subseteq KR$. However, when using our approach, the selected BK is one resource built from KR ontologies, called BBK. The BK selection computation-time is computed as follows.

- Traditional approach:

  $T_1 = \sum_{i=1}^{n} t(M, O_S, O_i) + \sum_{i=1}^{n} t(M, O_T, O_i) + \alpha.$

- Our approach:

  $T_1' = \sum_{i=1}^{n} t(M, O_S, O_i) + \beta.$

Where α and β are the computation times required for the treatments performed after the BK selection matching tasks. In the traditional approach, it may be the time of computing similarity measures and ranking the preselected ontologies (Hartung et al., 2012, Faria et al., 2014, Locoro et al., 2014). In our approach, it is the time of selecting the mappings related to the source ontology and combining them.

Usually, the values of α and β are negligible comparing to that of the matching tasks performed within the BK selection process.

In the example illustrated in Figure 4.4, with four preselected ontologies, the traditional approach performs eight matching tasks generating the alignments A_1 to A_8, while our approach performs four matching tasks generating the alignments A_1 to A_4.

For the anchoring step, we distinguish two cases:

Case 1: Reusing BK selection alignments as anchoring alignments.
The anchoring computation time is computed as follows.

- Traditional approach: the anchoring alignments are already available, no additional matching task is necessary.

  $T_2 = 0$.

- Our approach: The BK selection alignments are related only to the source ontology. Hence, another matching task is necessary to anchor the target ontology to the BBK (e.g., the time necessary to generate the alignment B_2 in Figure 4.4 (b)).

  $T'_2 = t(M, O_T, BBK)$.

The computation time of BK selection and anchoring is estimated as follows.

- Traditional approach:

$$T = T_1 + T_2 = \sum_{i=1}^{n} t(M, O_S, O_i) + \sum_{i=1}^{n} t(M, O_T, O_i) + \alpha. \tag{4.1}$$

- Our approach:

$$T' = T'_1 + T'_2 = \sum_{i=1}^{n} t(M, O_S, O_i) + \beta + t(M, O_T, BBK). \tag{4.2}$$

- Comparison: Traditional approach vs. our approach

$$T - T' = \sum_{i=1}^{n} t(M, O_T, O_i) - t(M, O_T, BBK) + (\alpha - \beta). \tag{4.3}$$

Intuitively the difference $T - T'$ is always positive. Indeed, the difference $(\alpha - \beta)$ tends to zero, and matching the target ontology to all the preselected ontologies takes more much time than matching the target ontology to the BBK. This intuition is validated with experiments in Section 4.6.2.

Case 2: The source and target ontologies are anchored to the selected BK with another matcher M'.

In our approach, the anchoring step requires two matching tasks, while in the traditional approach, the number of matching tasks depends on the number of the selected BK ontologies. For instance, in Figure 4.4 (a), with two selected BK ontologies, four matching tasks are necessary to generate B_1 to B_4. Thus, the anchoring computation time is computed as follows.

- Traditional approach:
  $T_2 = \sum_{j=1}^{k} t(M', O_S, SO_j) + \sum_{j=1}^{k} t(M', O_T, SO_j)$.
- Our approach:
  $T'_2 = t(M', O_S, BBK) + t(M', O_T, BBK)$.

The computation time of BK selection and anchoring is estimated as follows.

- Traditional approach:
  $T = T_1 + T_2 =$
  $(4.1) + \sum_{j=1}^{k} t(M', O_S, SO_j) + \sum_{j=1}^{k} t(M', O_T, SO_j)$.
- Our approach:
  $T' = T'_1 + T'_2 = (4.2) + t(M', O_S, BBK)$.
- Comparison: Traditional approach vs. our approach
  $T - T' = (4.3) + \sum_{j=1}^{k} t(M', O_S, SO_j) +$
  $\sum_{j=1}^{k} t(M', O_T, SO_j) - t(M', O_S, BBK)$.

Our hypothesis is that the difference $T - T'$ is always positive. Indeed, the formula (4.3) is positive as explained in **Case 1**, and matching the ontologies to align to the selected BK ontologies (i.e., $\sum_{j=1}^{k} t(M', O_S, SO_j) + \sum_{j=1}^{k} t(M', O_T, SO_j)$) takes more time than matching the source ontology to the BBK (i.e., $t(M', O_S, BBK)$). Note that, in our approach, matching the target ontology to the BBK (e.g., generating B_2 in Figure 8 (b)) is common to the two cases, and its computation time is already included in the formula (4.3). We discussed this case at the end of Section 4.6.2.

4.4 Experiment materials

4.4.1 Evaluation datasets

To evaluate our approach, we chose two OAEI tracks: Anatomy and Large biomedical ontology (LargeBio) that we have presented in Chapter 3. Our choice was motivated by the fact that, only for these tracks, state-of-the-art systems use ontologies

as background knowledge to enhance the quality of their alignments. Hence, evaluating with these tracks with the same preselected ontologies allows us to compare our results to the state-of-the-art ones.

4.4.2 Preselected ontologies

According to the OAEI 2016 campaign, AML (Faria et al., 2016) and LogMapBio (Jiménez-Ruiz et al., 2016) are the best BK-based ontology matching systems. To establish a fair comparison with these systems, our evaluation employs the same set of preselected ontologies as follows:

- **AML-Ontologies**: Three ontologies are preselected for AML: UBERON, DOID and MeSH[1]. AML makes a dynamic selection from these ontologies using the *Mapping Gain* measure (Faria et al., 2014).
- **LogMapBio-Ontologies**: In OAEI2016, LogMapBio considered the NCBO BioPortal as the set of preselected ontologies. LogMapBio selected 10 ontologies for each matching task. For our evaluation, we considered the combination of all the ontologies selected by LogMapBio as the preselected ontologies in order to establish a fair final result comparison. The combination yields 21 ontologies. However, YAM++ could not parse three of those ontologies. Indeed, these ontologies require importing external ontologies, a process which is not managed by YAM++. Thus, we ended up using 18 (out of the 21) ontologies for our comparison with LogMapBio. These ontologies are listed in Table 4.1 with their NCBO BioPortal acronyms[2]. Excluded ontologies are tagged by *.

For each matching task, we name **BBK1** the background knowledge resource built from AML-Ontologies and **BBK2** the one built from LogMapBio-Ontologies. Building the BK is performed according to the process described in Section 4.1 with YAM++ as a matcher.

4.4.3 Tools and resources

YAM++. YAM++ is an ontology matching system previously developed by our team at LIRMM[3] (Ngo and Bellahsene, 2016); it does not rely on a specialized BK to match biomedical ontologies. It is considered as one of the state-of-the-art ontology matching systems, and was the top ranked system in OAEI 2013. YAM++ combines several syntactic and structural similarity measures.

OBO DbXref. In addition to the mappings generated by YAM++, we also extracted cross-reference properties from the preselected ontologies when available

[1]MeSH is used as lexicon (Faria et al., 2015)

[2]These ontologies are accessible on NCBO BioPortal with the link https://bioportal.bioontology.org/ontologies/ontologyAcronym

[3]http://www.lirmm.fr/yam-plus-plus

Table 4.1: LogMapBio-Ontologies.

N°	Ontology acronym	Number of concepts
1	BIRNLEX	3,580
2	BTO	5,902
3	CCONT	19,991
4	CL*	2,352
5	CLO	40,884
6	CSEO	20,085
7	DDO*	6,444
8	DINTO*	28,178
9	DOID	12,432
10	EFO	19,909
11	EHDAA2	2,772
12	GALEN	23,141
13	HP	15,804
14	MA	3,257
15	ONTOAD	5,899
16	RCTV2	88,854
17	SYN	14,462
18	UBERON	19,761
19	VHOG	1,185
20	XAO	1,621
21	ZFA	3,168

(i.e., from the preselected ontologies present in the OBO Foundry). As previously pointed out (see Section 4.2.2), these cross-references may be considered as manually curated mappings. Therefore, we added them to the extracted mappings and assigned them a score of 1. Figure 4.5 shows an example of the OBO DbXref property; the concept of the UBERON ontology that has 0010501 as identifier, references the concept of the FMA ontology that has the identifier 72059. Hence, we ca extract the mapping $< UBERON : 0010501, FMA : 72059, \equiv, 1 >$

Neo4j. It is a graph database[4] intrinsically designed to work with paths within graphs. It implements many graph algorithms. Indeed, with relational databases, one has to implement an algorithm and perform several queries to find all paths between a given source and target concepts. Instead, with a graph database, a single simple query is sufficient.

Machine specifications. We run our experiments on an HP ZBook computer that has an Intel Core i7-4910MQ processor, 2.90 GHz of clock, 32 GB of RAM,

[4]https://neo4j.com/

```
<owl:Class rdf:about="http://purl.obolibrary.org/obo/UBERON_0010501">
    <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
        pseudostratified smooth columnar epithelium
    </rdfs:label>
    <rdfs:subClassOf rdf:resource="http://purl.obolibrary.org/obo/UBERON_0010498"/>
    <oboInOwl:hasDbXref rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
        FMA:72059
    </oboInOwl:hasDbXref>
    <oboInOwl:id rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
        UBERON:0010501
    </oboInOwl:id>
    <oboInOwl:hasRelatedSynonym rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
        non-ciliated pseudostratified columnar epithelia
    </oboInOwl:hasRelatedSynonym>
    <oboInOwl:hasOBONamespace rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
        uberon
    </oboInOwl:hasOBONamespace>
</owl:Class>
```

Figure 4.5: OBO DbXref example.

and a 64-bit Operating System (Windows 8.1 pro).

4.5 Implementation

In this section we will describe the technical implementation details of our BK selection approach.

Mapping extraction: We matched the preselected ontologies (i.e., AML-Ontologies or LogMapBio-Ontologies) between each others using the YAM++ matcher. For each matching task, YAM++ generates an RDF file that contains the mappings found between the input ontologies. YAM++ produces only equivalence mappings. Note that this process is performed once for each set of preselected ontologies, because it does not depend on the ontologies to align.

Mapping filtering: We matched the source ontology to the preselected ontologies using the YAM++. We then selected the mappings related to the source ontology according to the procedure described in Section 4.2.3. Ontology matching tools take as input two ontologies in OWL or RDF formats, and as we will see in the next chapter, the target ontology has to be matched to the built BK. Therefore, the implemented mapping filtering method generates two files: (i) an OWL file containing all the selected concepts with their labels; it may be seen as a fictional ontology that is created to enable anchoring to the target ontology by matching systems, and (ii) a CSV file containing all the filtered mappings in the following format (URI source, URI ontology source, URI target, URI ontology target, score, relation, manualMapping). *manualMapping* is a boolean property that takes "true" or "false" as value.

To generate the OWL file, we group the concepts per ontology, then for each on-

tology we perform two SPARQL queries: the first query is to extract the preferred labels of the concepts, while the second one is to extract the synonyms. We used the Jena API to load and query ontologies, and the SKOS vocabulary to describe the concept labels. Figure 4.6 shows an example of the description of two concepts belonging to two different ontologies (i.e., DOID and UBERON ontologies) in the OWL file.

```
<owl:Class rdf:about="http://purl.obolibrary.org/obo/DOID_9507">
   <skos:altLabel>ethmoiditis</skos:altLabel>
   <skos:altLabel>ethmoidal sinusitis</skos:altLabel>
   <skos:prefLabel>ethmoid sinusitis</skos:prefLabel>
 </owl:Class>
 <owl:Class rdf:about="http://purl.obolibrary.org/obo/UBERON_0003554">
   <skos:altLabel>rhombencephalon pia mater</skos:altLabel>
   <skos:altLabel>pia mater of hindbrain</skos:altLabel>
   <skos:altLabel>pia mater of neuraxis of hindbrain</skos:altLabel>
   <skos:altLabel>hindbrain pia mater of neuraxis</skos:altLabel>
   <skos:prefLabel>hindbrain pia mater</skos:prefLabel>
 </owl:Class>
```

Figure 4.6: Description of two concepts in the OWL file.

Mapping combination: In this step, we load the filtered mappings from the CSV file into Neo4j as a graph database. We used the command merge to ensure that each concept (node) is created once, hence linking the mappings that have common concepts.

4.6 Experimental evaluation

In this section, we evaluate the efficiency of our BK selection approach. We organize the evaluation in two sections. Each section is introduced with an assumption that we try to validate through experiments.

4.6.1 Built BK size vs. preselected ontologies size

***Assumption 1**: Our BK selection method builds a reduced-size BK comparing to that of the preselected ontologies.*

As discussed previously, our BK selection approach does not return a set of ontologies. Instead, it builds a novel knowledge resource that combines concepts selected from the initial preselected ontologies. To verify the assumption of this section, we compare the size (i.e., the number of concepts) of our built BK with that of the preselected ontologies (i.e., AML-Ontologies and LogMapBio-Ontologies).

In Table 4.2, for each matching task, we present the size of the built BK (BBK1 or BBK2) in number of concepts. Furthermore, we compute a percentage by dividing

the size of the built BK by the size of the preselected ontologies. BBK1 is built from three ontologies, which have a global size of 297,031 concepts while BBK2 is built from 18 ontologies, which have a global size of 302,707 concepts. For instance, the size of Task 1 BBK1 is 6,809; dividing 6,809 by 297,031 gives a percentage of 2%, which means that the BBK1 size represents only 2% of the preselected ontologies size.

The results reported in Table 4.2 validate Assumption 1. Indeed, for all matching tasks, the size of the built BK is much smaller than the size of the preselected ontologies. The percentage varies from one task to another with respect to the size of the ontologies to align. Tasks 2 and 6 share exactly the same built BK because they have the same source ontology; this shows that, when matching the source ontology with several target ontologies, the BK selection step may be performed once, and the built BK can be reused for each target ontology.

Table 4.2: Size comparisons: built BK vs. preselected ontologies.

Task	BBK1 size		BBK2 size	
Anatomy	3,173	1%	11,090	4%
Task 1	6,809	2%	18,104	5%
Task 2	46,280	15%	48,521	16%
Task 3	13,036	4%	27,465	8%
Task 4	16,251	5%	34,626	10%
Task 5	12,895	4%	36,456	12%
Task 6	46,280	15%	48,521	16%

4.6.2 Efficiency gain with the built BK

Assumption 2: *Our BK selection approach is more efficient than that of the traditional approach.*

Our approach reduces the computation time of the BK-based matching process, especially that of BK selection and anchoring as explained in Section 4.3. In our evaluation, we used the same matcher (i.e., YAM++) for BK selection and anchoring. Hence, we are in **Case 1** that reuses the BK selection alignments as anchoring alignments. We compare T and T' computed according to the formulas (4.1) and (4.2) introduced in Section 4.3, respectively.

In Figures 4.7 and 4.8, we present the following values:

- T: the time necessary for matching the source and target ontologies to the preselected ontologies in the traditional approach. We ignore α because it has a small value and variates from one work to another as explained in Section 4.3;
- T': the time necessary for mapping filtering, mapping combination and anchoring the target ontology to the built BK in our approach;

- the percentage ratio comparing the two values T and T'. This ratio is computed by dividing T' by T.

As we can observe, in all cases, our approach is more efficient than the traditional approach (i.e., $T' < T$). The gain is between 42% (for Task 2 with BBK1) and 60% (for Task 4 with BBK2). These results are expected since T and T' have a common part: matching the source ontology to the preselected ontologies. However, matching the target ontology to the preselected ontologies takes more time comparing to matching the target ontology to the BBK. For instance, in all tasks, matching the target ontology to BBK1 takes less than four minutes, while matching the target ontology to the large ontology MeSH always takes about 30 minutes.

With YAM++, the average time to match an ontology of LargeBio or Anatomy to: (i) one of the 18 ontologies listed in Table 4.1 is 2.8(min), (ii) the BK built from the 18 ontologies is 5.5(min). We may use these values to check our intuition about the efficiency gain in **Case 2**. When selecting only one ontology as background knowledge from the 18 ontologies, the difference between T and T' in Case 2 is computed as follows.
$T - T' = E + (2.8 + 2.8) - 5.5 = E + 0.1(min)$ where E is the value of the formula (3) that is the difference between T and T' in Case 1. As we can see in Figure 4.7 and 4.8, E is always positive. Hence, $E + 0.1(min)$ is positive.

In Case 2, the efficiency gain becomes more significant as the number of selected ontologies increases. For instance, with two ontologies as BK, the difference becomes:
$T - T' = E + (2.8 * 2 + 2.8 * 2) - 5.5 = E + 5.7(min)$

Based on the obtained results, we conclude that our BK selection approach builds an efficient BK, which validates Assumption 2. Indeed, the built BK reduces the BK selection and anchoring computation-time comparing to the traditional approach.

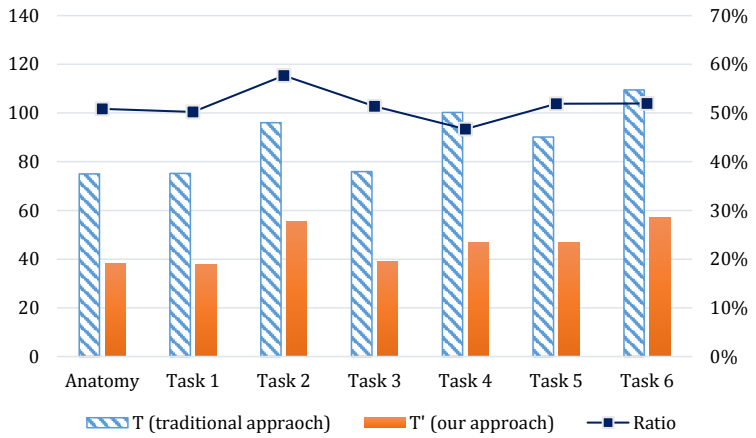

Figure 4.7: Efficiency gain with BBK1.

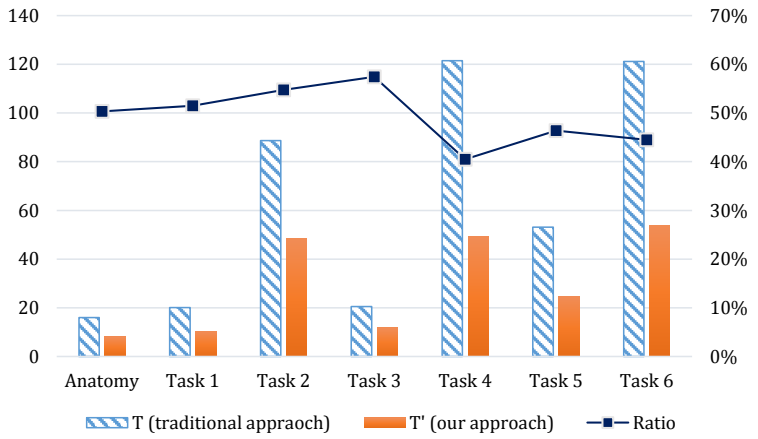

Figure 4.8: Efficiency gain with BBK2.

4.7 Conclusion

Ontologies are the knowledge resources the most used as background knowledge resources. However, a large number of ontologies is available which makes the BK selection a challenging issue. In this chapter, we have presented and evaluated our BK selection approach. Instead of returning a set of complete ontologies, our approach builds a novel knowledge resource from the initial preselected ontologies.

According to the experiments that we have conducted, our BK selection approach is efficient; it allows to reduce the BK selection time up to 60%.

As we have highlighted in Chapter 2, currently there is no measure to evaluate the effectiveness of a given BK for a given matching task. The idea is to exploit the selected BK and measure whether it allows to improve the recall of the direct matching alignment. In the next chapter, we will present our BK exploitation approach and verify the effectiveness of the built BK.

Chapter

5 BK Exploitation

Contents

5.1 Introduction

In this chapter, we present how we exploit the BK built in the previous chapter to derive mappings between the source and target ontologies. Related works showed that deriving mappings across several intermediate ontologies is more effective, however combining ontologies within the derivation process is time consuming (Sabou et al., 2008). As we will show, the built BK keeps the advantage of deriving across several intermediate ontologies without affecting efficiency.

Using background knowledge resources in ontology matching is a double-edged sword. Indeed, though these resources provide new information to find correct mappings, incorrect mappings may also be generated (Locoro et al., 2014). Consequently, selecting correct mappings from the candidate ones is particularly challenging in the context of BK-based matching. In this chapter, we propose two new selection methods. The first one is based on a set of rules, while the second one is based on supervised machine learning. To enable the use of a classification machine learning algorithm, we designed a set of 27 attributes based on the built background knowledge resource.

We performed extensive experiments on two datasets taken from OAEI, with two sets of preselected ontologies, to evaluate the performance of our approach. The experiment results confirm the effectiveness of our approach. Moreover, we compared our results to state-of-the-art systems that exploit background knowledge resources. Our F-measure values are very competitive relative to the best ones reported in the literature (Annane et al., 2018).

The rest of this chapter is organized as follows. Section 5.2 describes our BK exploitation approach. Then, Section 5.3 presents and analyzes the evaluation results and Section 5.4 discusses some limitations. Finally, Section 5.5 concludes this chapter.

5.2 Description of our BK exploitation approach

In this section, we present the different steps of our BK exploitation process, which includes:(i) anchoring, (ii) deriving candidate mappings and (iii) final mapping selection (see Figure 5.1).

5.2.1 Anchoring

Anchoring consists in identifying the entities of the ontologies to align in the background knowledge resource (Aleksovski et al., 2006b, Sabou et al., 2008). In our case, this is done by a direct matching between the ontologies to align and the built BK. Anchoring mappings are then added to the built BK graph. For more detail about the anchoring process, please see Section 2.3.3.1.

Note that, for the source ontology, we may simply reuse the mappings produced in the mapping filtering step (Section 4.2.3) between the source ontology and the

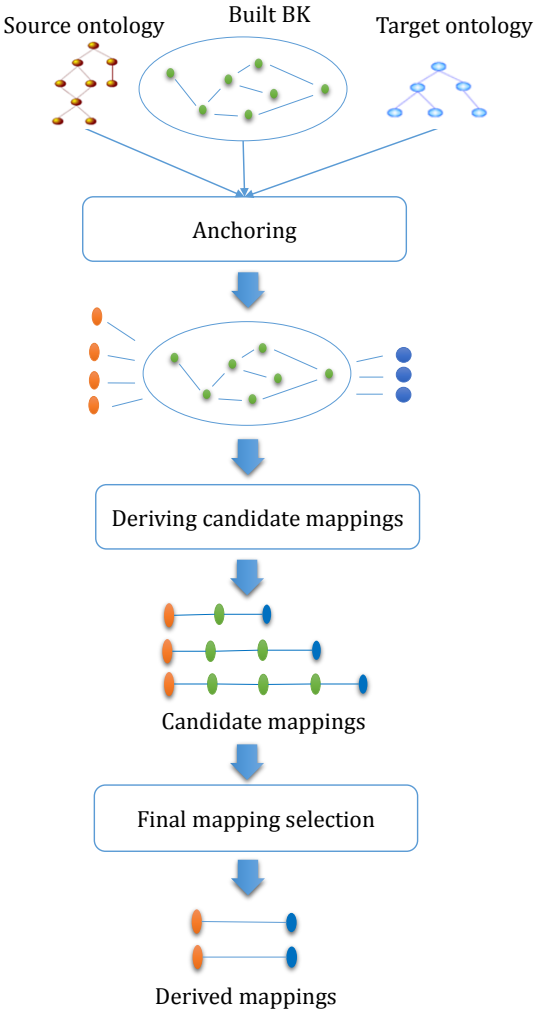

Figure 5.1: Overview of BK exploitation process.

preselected ontologies. This is feasible when both steps use the same matcher. However, BK selection and BK exploitation can be two completely independent steps.

5.2.2 Deriving candidate mappings

In this step, candidate mappings are derived between the ontologies to align using the graph structure of the built BK. We search for each source concept anchored to the built BK, all paths leading to the target ontology concepts. Each path found may be represented by a set of n mappings as follows:
$P = \{\langle e_{s1}, e'_{t1}, r_1, k_1\rangle, \langle e'_{s2}, e'_{t2}, r_2, k_2\rangle, ..., \langle e'_{sn}, e_{tn}, r_n, k_n\rangle\}$

Where e_{s1} belongs to the source ontology, e_{tn} belongs to the target ontology and $e'_{ti} = e'_{s(i+1)}$. Each path found provides a candidate mapping $\langle e_{s1}, e_{tn}, r, k\rangle$. r results from the composition of the different r_i on the path P. Similarly, k results

from the composition of the different k_i on the path P. In this thesis, we only deal with equivalence mappings (i.e., all r_i are equivalences). However, our approach may be extended to other kinds of mapping relationships, provided a strategy to compose different relationships on the same path is defined (Euzenat, 2008). Note that the intermediate concepts of a given path originate from different ontologies, which represents a derivation across several intermediate ontologies.

In ontology matching, the objective of using background knowledge resources is to complement direct matching but not to replace it. Indeed, direct matching may identify mappings that can be missed in BK-based matching and vice versa. Therefore, to complement the set of candidate mappings, we propose to add mappings resulting from the direct-matching between source and target ontologies to the set of derived candidate mappings.

5.2.3 Final mapping selection

To select the most accurate mappings, an effective mapping selection method must be used. Candidate mappings consist in a set of paths linking the source to the target concepts. Several paths may represent the same candidate mapping. Thus, to compute the final score k for a given candidate mapping, we must address two issues:

1. How to **compose** the different mapping scores of the same path?

2. How to **aggregate** the scores of different paths representing the same candidate mapping?

Related work suggested to use algebraic functions, such as multiplication, average, maximum, etc. to compose different mapping scores (Mascardi et al., 2010). These functions may also be used for aggregation (issue 2).

In the following, we use the term *configuration* for a given pair of composition and aggregation functions. For instance, the multiplication-maximum configuration means that the composition (issue 1) is the multiplication of the path scores, while the aggregation (issue 2) is performed with the maximum function. For a given candidate mapping, we may compute one or multiple scores according to the selection method. Indeed, different configurations return different scores for the same candidate mapping.

5.2.3.1 Rule-based selection

Rule-based selection of the final mappings consists in defining a set of rules to decide whether or not to keep a given candidate mapping in the final alignment. In our method, we propose the following rules:

1. Mappings returned by direct and indirect matching are selected.

2. Mappings resulting from the composition of only manual mappings are selected.

3. For each source concept, the target candidate with the highest mapping score is selected.

4. For each target concept, the source candidate with the highest mapping score is selected.

For rules 3 and 4, the score may be controlled by a given threshold. The score of the candidate mappings is computed with the multiplication-maximum configuration.

5.2.3.2 Machine learning-based selection

As previously discussed, there exist multiple possible algebraic function configurations to compose mapping scores of the same path, and to aggregate scores of different paths representing the same candidate mapping. However, testing the performance of all possible configurations to find the most suitable one for a given matching task is fastidious. Additionally, manually finding the best configuration for a given matching task does not amount to finding it for all matching tasks. Furthermore, one may combine several configurations to improve the effectiveness of the selection method; for example, one could combine average-multiplication, maximum-multiplication and average-average configurations. Indeed, each configuration may provide a piece of information which could help to select the most relevant mappings. In this case, however, we would also have to define how to combine the different values of these configurations to select the final mappings. This renders the task even more complex.

Supervised Machine Learning technique (ML) is an appropriate option to address this issue. Indeed, according to the training data, ML automatically customizes a classification function (classifier) that combines several attributes (selection variables). We therefore propose to cast the problem of mapping selection into a classification problem as follows:

- The test data are the candidate mappings between the source and target ontologies to be classified as *true* or $false$.

- The training data are a set of candidate mappings already classified as *true* or $false$. These candidate mappings are completely distinct from the test data (the candidate mappings to classify).

- The attributes (or features) that describe each candidate mapping are the different configurations and any variable that can help to classify a given candidate mapping.

In the following, we present the candidate mapping attributes, the training data as well as *RandomForest*, the machine learning algorithm used in this article.

5.2.3.3 Candidate mapping attributes

In our case, the attributes are the selection variables. Indeed, each attribute is a decision variable that will help to decide if a given candidate mapping will be classified as *true* or $false$. In related work, to classify the candidate mappings, similarity measures between source and target concepts were used. Here, however, the candidate mappings are a set of paths between source and target concepts. Therefore, we need to define new attributes. We thus propose a set of 27 selection attributes for each candidate mapping:

Direct score: if the candidate mapping belongs to the alignment returned by the direct matching, the direct score is the score of the candidate mapping in this alignment; otherwise, it is 0. Our intuition is that the mappings returned by the direct matching are likely to be correct.

Number of paths representing the candidate mapping: in fact, candidate mappings returned by many paths are more likely to be correct than those returned by few paths.

Path length attributes: for each candidate mapping, we compute three attributes that are (i) the minimum length, (ii) the maximum length and (iii) the average length of paths that represent the candidate mapping. Our intuition is that, the shorter the paths, the more relevant the candidate mapping will be.

Mapping score attributes: For each candidate mapping, 21 score attributes are computed. Indeed, for each path that represents the candidate mapping, we compute seven values with the following composition functions: (1) maximum, (2) minimum, (3) average, (4) multiplication, (5) sum, (6) variance and (7) average divided by variance. Each function takes the scores of the mappings that make this path as an input. We then aggregate path scores for each composition function with three functions: (1) maximum, (2) minimum and (3) average. For instance, when using variance as a composition function, we compute three attributes from the paths that represent the candidate mapping with the following configurations: maximum-variance, minimum-variance and average-variance. We repeat this process with the other six composition functions to obtain 21 attributes.

Maximum average of manual mappings: For each path representing the candidate mapping, we compute the average number of manual mappings (i.e., the number of manual mappings divided by the number of mappings of this path). Then, the maximum average is taken as an attribute. Indeed, paths containing manual mappings are more relevant than those containing only automatic mappings.

Let us take an example to illustrate the computation of the various attributes. Figure 5.2 shows an actual example from our evaluation (described further). Concepts are represented in the form: ontology#ConceptIdentifier; the values on edges are the mapping scores returned by the automatic matcher; OBO is a manual mapping. As we can see, the source concept is anchored to three BK concepts, while the target concept is anchored to only one. The derivation step returns four paths linking the source concept to the target concept.

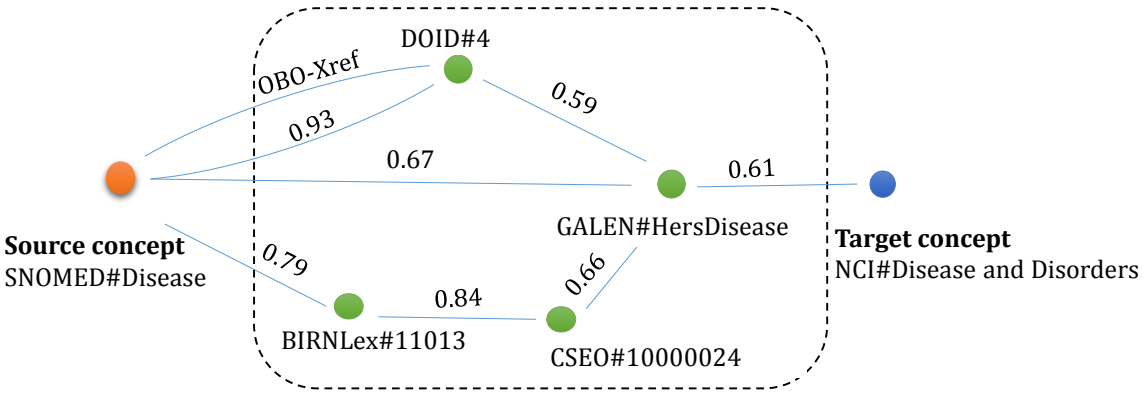

Figure 5.2: Example of candidate mapping derivation.

The following candidate mapping:
(SNOMED#Disease, NCI#Disease and Disorders) is described by the following attributes:

Direct mapping score: 0; number of paths: 4; Average path length:$(3 + 3 + 2 + 4)/4 = 3$; minimum path length: 2; maximum path length: 4; Average manual mapping: 1/3 because there is only one path of length 3 that contains one manual mapping; For score attributes, we illustrate one (multiplication) of the seven composition functions proposed. We start by computing a score for each path, as shown in Table 5.1. Then, using these path scores, we compute the following attributes: maximum scores: 0.41; minimum scores: 0.27; average scores: (0.36+0.33+0.41+0.27)/4=0.34.

Table 5.1: Path scores for Figure 5.2 example.

Path nodes	Score
DOID#4, GALEN#HersDisease	1*0.59*0.61=0.36
DOID#4, GALEN#HersDisease	0.93*0.59*0.61=0.33
GALEN#HersDisease	0.67*0.61=0.41
BIRNLex#11013, CSEO#10000024, GALEN#HersDisease	0.79*0.84*0.66*0.61= 0.27

5.2.3.4 Training data

In our case, training data are candidate mappings annotated by *true* (correct mapping) or *false* (incorrect mapping) and described by all the previously presented attributes. As is usual with supervised machine learning, obtaining training data

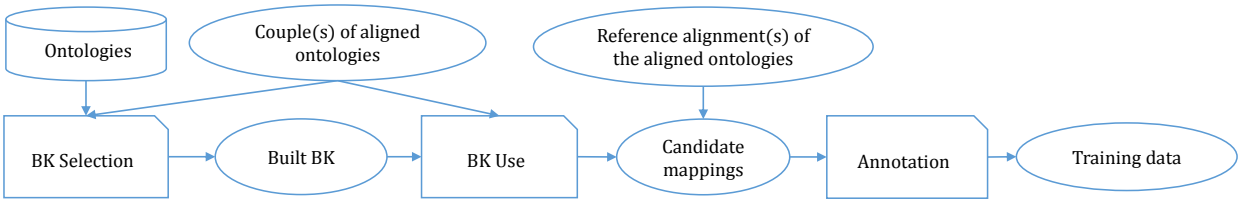

Figure 5.3: Training data generation process.

requires previously generated and curated reference alignments from other ontologies than those to align. Preferably, the aligned ontologies are of the same domain as the ontologies to align. To obtain the training data, we propose to apply our approach to the aligned ontologies (i.e., BK selection and BK exploitation). We then, compute the 27 attributes for each derived candidate mapping and annotate it by *true* or *false* according to the reference alignments of the aligned ontologies (see Figure 5.3).

5.2.3.5 RandomForest machine learning algorithm

There are several algorithms for learning a classification function from a set of training data. In our experiments, we used *RandomForest*, a non-linear method for classification (Breiman, 2001). In the training step, it learns a multitude of decision trees by creating a different random subset to train each decision tree. In the classification step, it aggregates the results of these trees by outputting the most frequent class. Due to this strategy, Random Forest has the advantage of being effective on any type of dataset. Our choice of this algorithm was motivated by its performance in preliminary experiments. Indeed, we evaluated the classification results produced by different ML algorithms implemented in the Weka framework (Hall et al., 2009) such as trees algorithm (J48, RandomForest, RandomTree) and rules algorithms (JRIP, oneR, etc.); RandomForest generated the best results. This corroborates the results reported in (Isele and Bizer, 2012): for learning linkage rules, the non-linear classifiers (trees) are the most appropriate.

5.3 Experimental evaluation

In this section, we evaluate our BK-based ontology matching approach through several experiments. We organize the evaluation in four sections. Each section is introduced with an assumption that we try to validate through experiments. The experimental materials are the same described in the previous chapter.

5.3.1 Deriving mappings across several intermediate concepts

Assumption 1: *Deriving mappings across several intermediate concepts generates more correct mappings than deriving across one intermediate concept.*

In our approach, mapping derivation is performed by searching all paths between source and target concepts. Each path contains a number of intermediate concepts belonging to the preselected ontologies. For instance, with three preselected ontologies, we may derive mappings with paths that contain one intermediate concept, two intermediate concepts or three intermediate concepts. In our experiments, we derived mappings with a maximum of three intermediate concepts.

In Tables 5.2, 5.3 and 5.4, we present the number of correct and incorrect candidate-mappings, which have been derived across 1, 2 and 3 BK concepts. Then, to observe the total number of the correct and incorrect derived mappings, we computed the values *Sum correct* and *Sum incorrect*. For instance, in Table 5.2, for Task 1 and path length equals 3, the number of distinct correct mappings derived with paths with length 2 and 3 is 2013.

The small fragment tasks (i.e., Task 1, Task 3 and Task 5) have the same reference alignments as the large fragment tasks (i.e., Task 2, Task 4, and Task 6, respectively). Hence, the same number of correct candidate-mappings. They do not have the same number of incorrect mappings derived, but for the sake of readability, and since we focus in this section on studying the benefit of deriving across several intermediate concepts to discover correct mappings, we only present the results of the small fragment tasks in Tables 5.2 and 5.3.

As we can observe, in all cases, paths of various length return much more correct mappings than incorrect ones. We note that, generally, the incorrect mappings are less precise ones; we mean by less precise that the returned entities are related, but not equivalent, since the reference alignment contains only equivalent mappings, these *less precise* mappings are considered as incorrect.

Sum correct and *Sum incorrect* rows allow to study the complementarity of the various length paths to discover correct mappings. Increasing the number of intermediate concepts (i.e., BK concepts), discovers more correct mappings. The number of the new discovered correct candidate-mappings decreases from one path length to another. For Anatomy, few new correct mappings are discovered with more than one BK concept. This may be explained by the use of UBERON as BK ontology, which is an integrative multi-species anatomy ontology. Indeed, UBERON, employed as the only BK ontology, allows to identify more than 80% of Anatomy reference alignment mappings.

Table 5.5 summarizes the results of Tables 5.2, 5.3 and 5.4. It shows the following columns:

- **A**: the number of correct mappings derived across one BK concept;
- **B**: the number of correct mappings derived across one, two and three BK

Table 5.2: LargeBio: correct and incorrect candidate mappings using BBK1.

Task	#BK concepts	Correct	Incorrect	Sum correct	Sum Incorrect
Task 1	1	1938	80	1938	80
	2	1710	106	2013	124
	3	1509	119	**2054**	**147**
Task 3	1	2043	235	2043	235
	2	1481	264	2125	347
	3	958	276	**2158**	**386**
Task 5	1	4789	1298	4789	1298
	2	3725	1820	5038	2203
	3	3096	3178	**5091**	**3905**

Table 5.3: LargeBio: correct and incorrect candidate mappings using BBK2.

Task	#BK concepts	Correct	Incorrect	Sum correct	Sum Incorrect
Task 1	1	2369	118	2369	118
	2	2173	195	2423	205
	3	2217	237	**2442**	**247**
Task 3	1	2511	306	2511	306
	2	2045	464	2600	520
	3	1983	544	**2685**	**623**
Task 5	1	9871	2123	9871	2123
	2	6933	2811	10068	3101
	3	7170	4477	**10315**	**4818**

Table 5.4: Anatomy: correct and incorrect candidate mappings.

BK	#BK concepts	Correct	Incorrect	Sum correct	Sum Incorrect
BBK1	1	1401	93	1401	93
	2	1092	64	1404	110
	3	344	66	**1405**	**128**
BBK2	1	1411	103	1411	103
	2	1412	109	1418	119
	3	1246	137	**1420**	**152**

concepts;

- **Gain**: the percentage of gain when using several BK concepts (Gain = $\frac{B-A}{A}$).

As we can see, Assumption 1 is validated. Indeed, for each matching task, the derivation across several intermediate concepts generates more correct mappings, with a gain of up to 7%, compared to deriving mappings with only one intermediate concept. Note that deriving mappings with only one intermediate concept is comparable to deriving with each BK ontology separately from the other BK ontologies, which is the method adopted by almost all related works (Faria et al., 2016, Jiménez-Ruiz et al., 2015, Hartung et al., 2012, Quix et al., 2011). Instead, thanks to the mapping extraction task, our approach combines all preselected ontologies.

When deriving mappings across several intermediate concepts, we may notice that the number of correct mappings derived with BBK1 is comparable to BBK2 for Tasks 1, 2, 3 and 4. However, for Tasks 5 and 6, the gap is larger: 10,315 correct mappings are derived with BBK while only 5,091 correct mappings are derived with BBK1. This shows that BBK2 is more effective than BBK1 for these tasks.

Table 5.5: Evaluation of derivation effectiveness across several BK concepts.

BK	Task	A	B	**Gain**
BBK1	Anatomy	1,403	1,405	**0.1%**
	Task 1 & Task 2	1,938	2,054	**6.0%**
	Task 3 & Task 4	2,043	2,158	**5.6%**
	Task 5 & Task 6	4,789	5,091	**6.3%**
BBK2	Anatomy	1,411	1,420	**0.6%**
	Task 1 & Task 2	2,369	2,442	**3.1%**
	Task 3 & Task 4	2,511	2,685	**6.9%**
	Task 5 & Task 6	9,871	10,315	**4.5%**

5.3.2 Final mapping selection

Assumption 2: *Our rule-based and ML-based mapping-selection methods are effective.*

As previously discussed, the selection of the most relevant mappings from the candidate ones is a crucial step in BK-based ontology matching as exploiting BK leads to discover more correct and incorrect mappings. We have described two mapping selection methods in Section 5.2.3 that we evaluate here to validate Assumption 2. For our experiments, we have implemented the following mapping selection methods:

1. **Baseline.** This is the simplest method, it consists in keeping all candidate mappings that have been derived without any selection.

2. **Rule-based selection.** This method is the implementation the rules described in Section 5.2.3.1 to select the final mappings.

3. **ML-based selection.** To evaluate the ML selection, we have implemented two strategies that use the same ML algorithm and the same attributes to describe candidate mappings but generate the training data differently. Note that in both strategies there is a complete distinction between the test data (candidate mappings to classify) and the training data.

   (a) **Cross validation.** This strategy is often used to evaluate the performance of the ML algorithm and attributes used for a given matching task in case of using a training data objects similar to the objects to classify. The process for a given matching task is as follows: (1) we subdivide the set of candidate mappings of this task into two equal subsets. (2) We annotate the candidate of the first subset with *true* or $false$ according to the reference alignment. (3) We use the annotated subset as training data to learn a classifier. (4) Then, we classify the candidate mappings of the second subset with the resulted classifier. (5) We interchange the two subsets such as we annotate the second subset and classify the candidate of the first one. (6) Finally, we combine the two classification results (we take all candidate mappings classified as *true*) to obtain the final alignment.

   (b) **Separate learning.** Here, we generate the training data for a given matching task using the ontologies and reference alignments of other tasks. For LargeBio benchmark, we adopt a *leave one out* strategy. For each task, we generate the training data using the others tasks of the same size. For instance, we have three large fragments tasks (Task 2,Task 4 and Task 6), to classify the candidate mappings of Task 2, we use the ontologies and reference alignments of Task 4 and Task 6 to generate the training data according to the process illustrated in Figure 5.3. For Anatomy, we generate the training data with Task 1, Task 3 and Task 5.

Why 2-fold cross validation? We used the 2-fold cross validation rather than the 10-fold (the most used one) to show that for a given matching task, if we have a partial reference alignment (about 50%), we may use it to learn an effective classifier for selecting the most relevant mappings among the candidate ones for the rest of concepts. Indeed, the cross-validation strategy with 10-fold uses 90% of the annotated data as training data and the left 10% as test data. In this case we have more training data, hence a more effective classifier. Indeed, the results with 10-fold cross validation are slightly higher.

To fairly evaluate the performance of our selection methods, in this section, we compute the recall with respect to the number of correct mappings that could be derived, and not to the number of mappings in the reference alignment. Indeed, if some correct mappings are not available in the set of candidate mappings, we cannot blame the selection method for not having returned them.

$$Recall = \frac{TP}{TPG}$$

Where TP is the number of the correct mappings returned by a given selection method and TPG is the number of all correct mappings that could be derived using the built BK.

Figures 5.4, 5.5 and 5.6 present the results of our experiments for each matching task using respectively BBK1 and BBK2. In particular, we present the precision, recall and F-measure of the final alignment to observe the behavior of each mapping selection method.

Precision

As we can see in Figures 5.4 (a) and (b), the baseline's precision for small size tasks (Task 1, Task 3 and Anatomy) is comparable to that of other selection methods. However, for larger size tasks (Tasks 2, 4, 5 and 6), the precision is low, especially for Tasks 2 and 6.

Even if the precision curves display the same trend in Figures 5.4 (a) and (b), the scores in Figure (b) are lower than those in Figure (a). This may be explained by the fact that BBK2 is built from a larger number of preselected ontologies than BBK1 18 vs. 3 ontologies). Hence, BBK2 generates more correct (see Table 5.5) and incorrect mappings, which decreases precision.

The ML-based selection methods consistently yields higher precision than the rule-based selection method, with an average of 0.915 for cross-validation and 0.909 for separate learning (vs. 0.881 for the rule-based selection). The largest gap is observed in Task 2. This is due to the fact that the NCI Thesaurus includes a small branch on mouse anatomy in addition to the human anatomy branch. Using the cross-references extracted from UBERON (considered as manual mappings) and the selection rule number 2 (see Section 4.1), the rule-based selection method returns mappings between human and mouse anatomy. However, the UMLS, the source from which the reference alignment is extracted, is focused only on human health, and does not include mappings between the NCI mouse anatomy branch and MA (the Mouse Anatomy ontology); therefore, these mappings are considered as incorrect, which affects precision (Faria et al., 2014).

Recall

The baseline always shows a recall of 1, because we computed a customized recall as described above (see Figures 5.5 (c) and (d)). Our selection methods yield a high recall in all matching tasks. The rule-based mapping selection method obtained the best recall scores, with an average of 0.979, while the cross-validation and separate-learning methods had a recall average of 0.955 and 0.938, respectively.

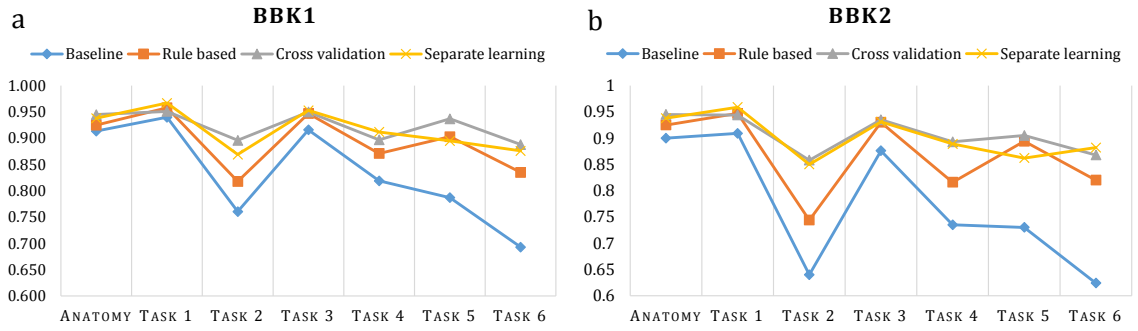

Figure 5.4: Selection method comparison: Precision.

The difference between the rule-based and separate-learning selection is significant in Task 4, while the gap is smaller with cross-validation. This may be explained by the low precision of the baseline alignment of Tasks 2 and 6. This affects the learned classifier. Indeed, the baseline alignments of Tasks 2 and 6 are the training data of Task 4. Training data contains many *false* candidate mappings increase the probability of classifying a given candidate mapping as *false*, which, in turn, decreases recall.

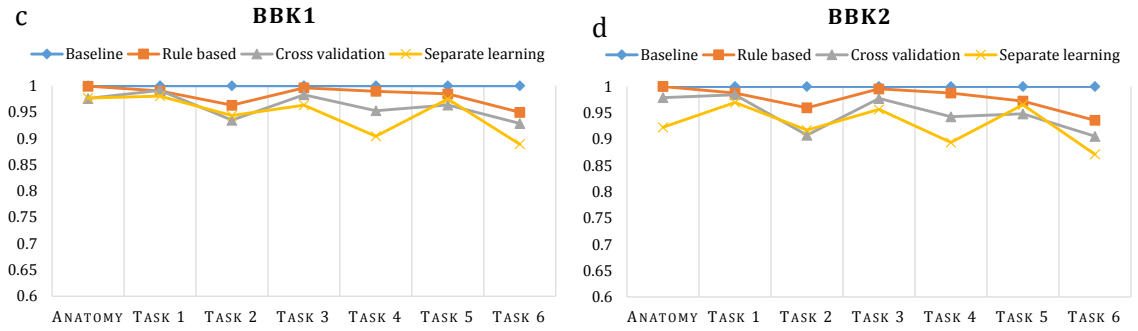

Figure 5.5: Selection method comparison: Recall.

F-measure

We present the F-measure values in Figures 5.6 (e) and (f). The cross-validation method yielded the best F-measure scores with an average of 0.942 when using BBK1 and of 0.928 when using BBK2. These results demonstrate that the ML technique with the proposed attributes and similar training data is effective for the mapping selection task. ML-based mapping selection is therefore particularly well-suited for complementing an existing partial alignment between two ontologies (the partial alignment may be used to generate the training data) (Lambrix and Liu, 2009, Mary et al., 2017), or for matching new ontology versions when an alignment between the old ontology versions already exist. Indeed, the training data may be generated with the existing alignment.

The separate-learning method produced high F-measure scores as well, close to the cross-validation method's scores, with an F-measure average of 0.931 and 0.914 when using BBK1 and BBK2, respectively. These results are more interesting. They show a concrete case where we may reuse existing alignments within the same domain to learn an effective classifier. Note that, we generated the training data for the Anatomy (that has a gold standard reference alignment) using alignments extracted automatically from UMLS (Tasks 1, 3 and 5 reference alignments), and the selection results are promising.

The rule-based method provides results with an average of 0.931 and 0.910 when using BBK1 and BBK2, respectively. It obtained the best F-measure values for the small tasks (i.e., Anatomy, Tasks 1 and 3). However, its performance decreases (i.e., achieves lower precision) for large tasks, compared to ML-based selection, which is more stable.

The results of the ML-based and rule-based mapping selection methods are comparable in terms of F-measure scores. However, the ML-based selection promotes precision, while the rule-based selection promotes recall.

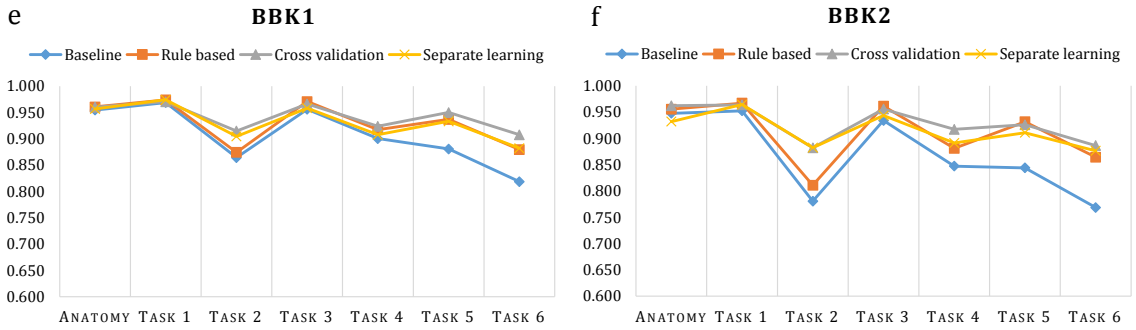

Figure 5.6: Selection method comparison: F-measure.

Discussion

Rule-based mapping selection is simple and efficient but static. Indeed, although each ontology-matching task has its own specificities (for instance, the best threshold value varies from one task to another), the same rules are applied all the time. ML-based mapping selection is time consuming and requires aligned ontologies to generate training data. However, it dynamically builds a customized classifier that combines multiple selection attributes (27 in our case). Mappings that are manually created or validated stored in platforms such as YAM++ online (Bellahsene et al., 2017)[1], NCBO BioPortal or resources such as OBO ontologies may be used to generate training data.

Based on our experiment results, we can validate Assumption 2. Our selection methods are effective: they significantly improve baseline precision and consistently keep high recall.

[1] http://yamplusplus.lirmm.fr/

5.3.3 Effectiveness of the built BK

Assumption 3: *The small size of the built BK does not affect its effectiveness.*

In this section, we evaluate the effectiveness of our approach by comparing its results to (i) the direct matching results which are the alignments generated with YAM++, and (ii) the indirect matching alignments generated with LogMapBio and AML within the OAEI 2016 campaign.

5.3.3.1 Our results vs. the direct matching results of YAM++

In Figures 5.7, 5.8 and 5.9, we present the final results of our approach with the different selection methods described in Section 5.3.2 (CV: cross validation, SL: separate learning, R: rule based selection). As we can see, our approach significantly improved the results of the direct matching performed with YAM++, mainly by increasing recall. For instance, for Anatomy, when exploiting BBK1 (see Figure 5.7), our approach increased the F-measure value from 0.841 to 0.929. This may be explained by the effectiveness of the built BK, which generated more correct mappings (high recall), and that of the mapping selection methods, which insured high precision too. These results legitimate the current trend of exploiting BK resources to enhance ontology matching.

When using BBK2, the direct matching with YAM++ has the best F-measure value for Task 2. This may be explained by the loss in precision because of using UBERON as explained previously in Section 5.3.2.

As expected (see Section 5.3.2), the F-measure values obtained using the various mapping selection methods are comparable however, and especially for large fragment tasks, the ML based methods have the best precision values while the rule based method has the best recall values.

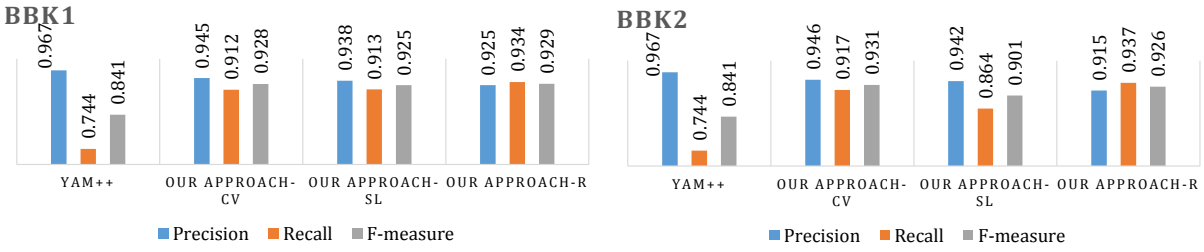

Figure 5.7: Anatomy results.

5.3.3.2 Our results vs. the state-of-the-art results

According to the OAEI campaigns (Achichi et al., 2016), AML and LogMapBio are the best systems using ontologies as background knowledge. Hence, comparing our

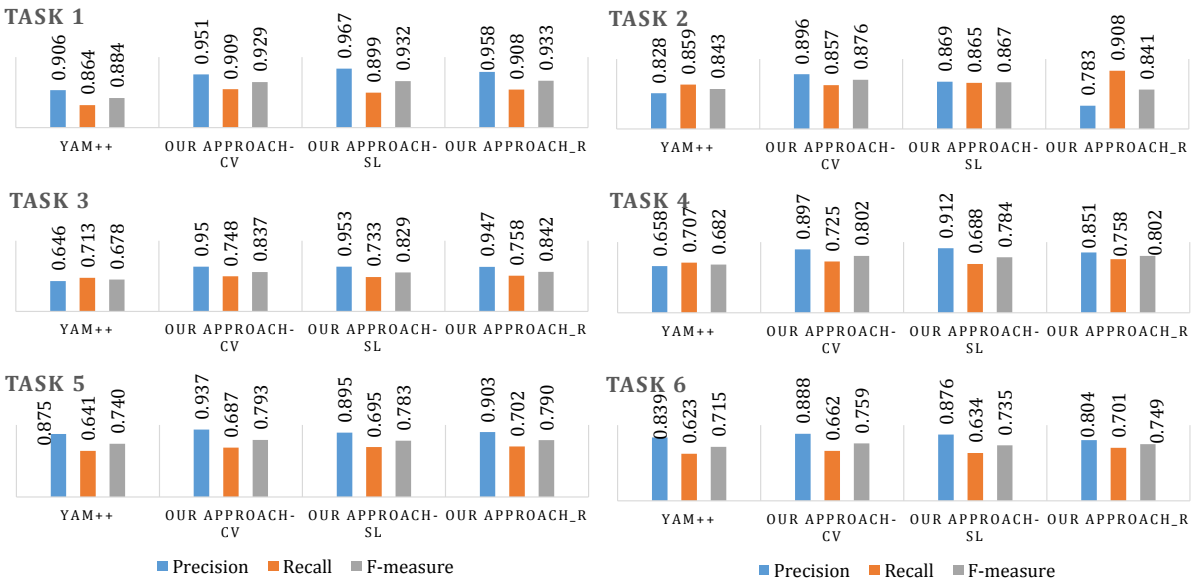

Figure 5.8: LargeBio results exploiting BBK1.

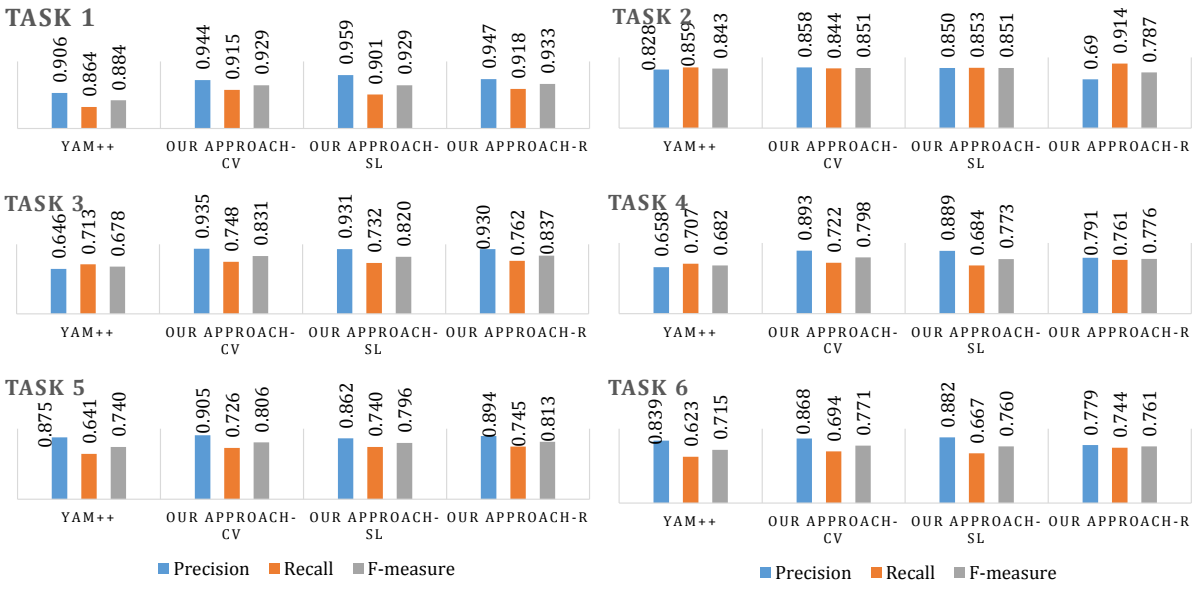

Figure 5.9: LargeBio results exploiting BBK2.

results to that of these systems permits to evaluate the effectiveness of our approach. For a fair comparison, (i) our evaluation uses the ontologies that were preselected for these systems in the OAEI 2016 campaign, (ii) only our rule-based selection results are compared since the OAEI rules prohibit training on OAEI datasets[2], and (iii) we repaired the alignments generated by our approach with the LogMap's ontology repair module (LogMap-Repair) (Jiménez-Ruiz et al., 2013), which is available as a self-contained software component[3]. Indeed, AML and LogMapBio use logical repair strategies to ensure the coherence of their alignments.

The aim of this comparison is to evaluate the performance of our approach regarding the best results obtained using the same preselected ontologies. Thus, if we obtain comparable results, we can conclude that the reduced size of the built BK does not affect its effectiveness.

In Table 5.6, we present the difference between the F-measure values of the repaired alignments and those of the original ones. Generally, repairing the alignments with LogMap-Repair has a positive impact on the F-measure values. This impact is more significant when using BBK2, especially for Task 2 and 4. This may be explained by the fact that BBK2 generates more incorrect mappings than BBK1 (see Tables 5.2, 5.3 and 5.4; the values in bold are the baseline values), hence more incoherent mappings.

Table 5.6: Repairing gain with LogMap-Repair.

Task	Gain (BBK1)	Gain (BBK2)
Anatomy	0.003	0.003
Task 1	-0.001	0.001
Task 2	0.007	0.034
Task 3	0.001	0.003
Task 4	0.002	0.015
Task 5	0.001	0.001
Task 6	0.004	0.008

In Figures 5.10 and 5.11, we present the precision, recall and F-measure values of the alignments returned by:

- AML when our approach exploits BBK1.
- LogMapBio when our approach exploits BBK2.
- Our approach with the rule-based mapping selection method, and the LogMap-Repair module.

The results of AML and LogMapBio are those reported in the OAEI 2016 campaign.

[2]http://oaei.ontologymatching.org/doc/oaei-rules.2.html
[3]https://code.google.com/archive/p/logmap-matcher/downloads

Our approach slightly overcomes AML results in three tasks (Task 1,3 and 4) and has a close results for the other tasks (see Figure 5.10). We may say that our results are comparable to AML results in case of using the three preselected ontologies. Our approach has a higher recall, while AML has a higher precision, which leads to the same F-measure average of 0.843.

Our approach outperforms LogMapBio results in all tasks except Task 2 (see Figure 5.11). This may be the result of the derivation across several intermediate concepts that increases the recall and the F-measure of our results. Indeed, LogMapBio composes only two mappings related to the same BK ontology at a time.

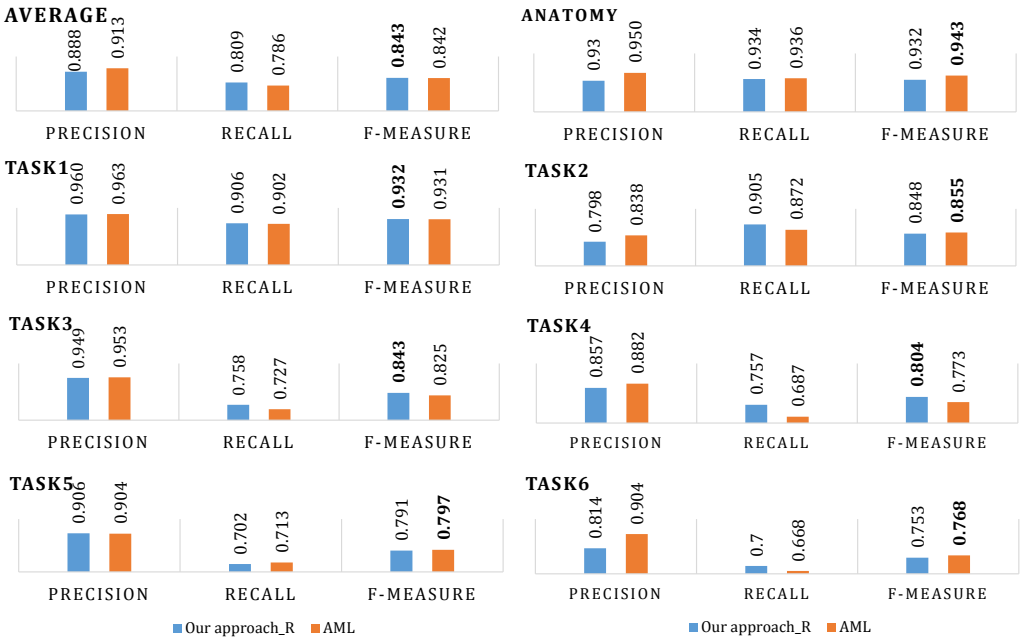

Figure 5.10: Result comparison: our approach exploiting BBK1 vs. AML.

The use of the built BK increases the recall and more generally the quality (F-measure) of the direct matching alignment. In addition, it provides results very competitive to the state-of-the-art results, which corroborates Assumption 3.

5.3.4 Computation time evaluation: step by step

Assumption 4: *The use of ontologies as background knowledge has a computation time cost and our approach reduces this cost.*

To validate Assumption 4, we evaluate the time necessary to perform the different steps of our approach. To ensure a global evaluation, we include the computation time of the BK selection step (see Chapter 4).

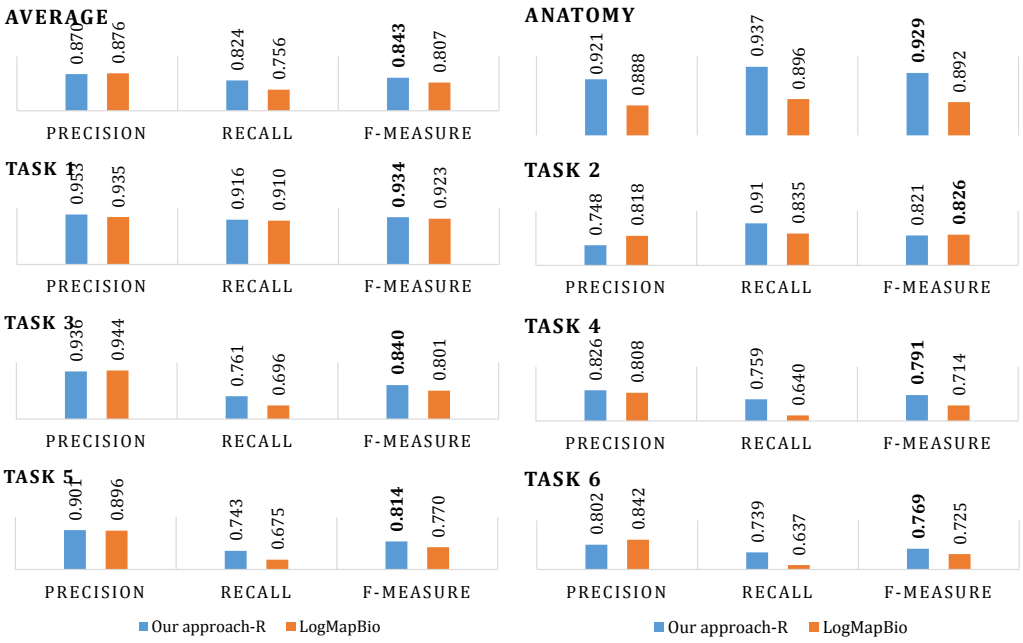

Figure 5.11: Result comparison: our approach exploiting BBK2 vs. LogMapBio.

Figures 5.12 and 5.13 show the time, in minutes, required for the different steps in our approach.

BK selection

The BK selection step includes three tasks: mapping extraction, mapping filtering and mapping combination.

Mapping extraction is the costliest task in terms of computation time, especially when using a large number of preselected ontologies, as is the case for BBK2 (18 ontologies). Indeed, extracting the mappings from the AML-Ontologies and LogMapBio-Ontologies took 77 and 132 minutes, respectively. Fortunately, this task is performed independently of matching tasks. In fact, for a given set (or repository) of preselected ontologies, the mapping extraction task is performed only once whereas its output (the set of alignments) is reused for any matching task. Therefore, we report the computation time of the mapping extraction process once for all matching tasks in Figures 5.12 and 5.13.

BBK1 is built from three ontologies. However, the time necessary for extracting the mappings from the three ontologies is 58% the time necessary for performing the same process from 18 ontologies. This may be explained by the fact that, in terms of computation time, matching a large preselected ontology such as MeSH is equivalent to matching several small ontologies.

The high computation time cost of the mapping extraction step is justified by the fact that the mapping derivation across several intermediate concepts generates more correct mappings, as demonstrated in Section 5.3.1.

Mapping filtering is the second costliest process. It includes two tasks: (i) match-

ing the source ontology to the preselected ontologies and (ii) selecting the mappings related to the source ontology. The first task is time-consuming, especially when dealing with large ontologies such as MeSH. Indeed, it is surprising to notice that matching the source ontology to 3 preselected ontologies takes more time than matching it to 18 preselected ontologies (see Figures 5.12 and 5.13). MeSH contains 265,414 concepts, and each concept is described with multiple labels. Therefore, YAM++ takes long time to match MeSH with any ontology, particularly when the latter is large too. The second task takes only few seconds in all cases.

Mapping combination is performed with Neo4j allowing us to merge the same nodes of different mappings. It takes less than 2 seconds in all cases.

Anchoring and derivation

These steps take much less time than the BK selection step. The size of the target ontology is larger than that of the source ontology however, we notice that anchoring the target ontology takes much less time than matching the source ontology to the preselected ontologies in the mapping filtering step. This may be explained by the fact that the target ontology is anchored only to the reduced-size built BK.

The derivation task is performed with Neo4j. It takes less than one minute for small matching tasks and up to three minutes for large ones.

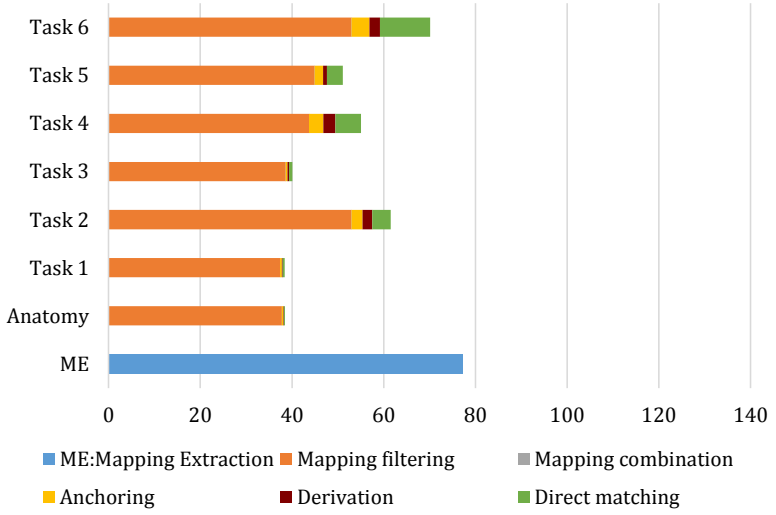

Figure 5.12: Computation time in minutes (BBK1).

Final mapping selection

The rule-based mapping selection method takes less than two seconds in all cases. However, when using ML-based selection, we have to precise what we consider as the computation time of the mapping selection process. Indeed, classifying the candidate mappings into true or false takes only few seconds however, generating the training data and building the classifier are time consuming. For example, in our evaluation, we used Tasks 1 and 3 to generate the training data for classifying Task 5 candidate mappings. Hence, the time necessary to generate the training data in this case is the

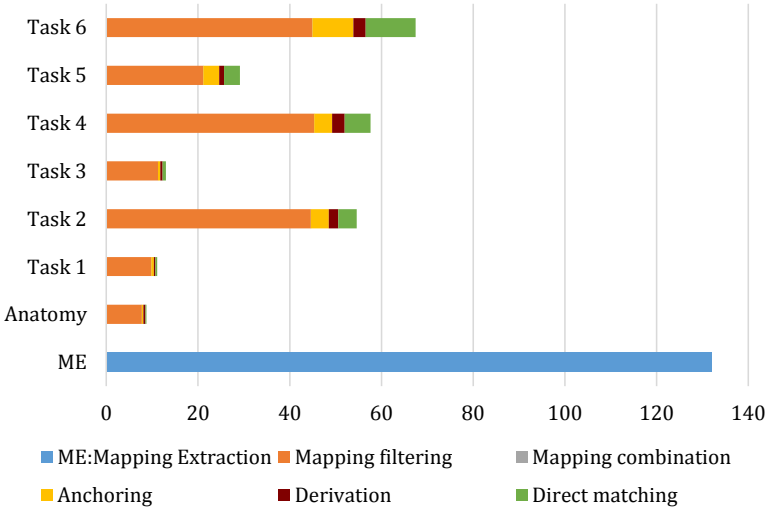

Figure 5.13: Computation time in minutes (BBK2).

time necessary for BK selection and BK exploitation of Tasks 1 and 3. Note that the training data for Task 5 may be generated with Task 1 only. Furthermore, the learned classifier is reusable in the same domain. Indeed, we tried to classify Task 4 candidate mappings derived from BBK2 with the classifier trained with Tasks 2 and 6 candidate mappings derived from BBK1. We obtained almost the same results as those obtained with the classifier trained with candidate mappings derived from BBK2. Hence, spending time to learn one classifier for a given domain is acceptable since it can be reused for different matching tasks.

The obtained results validates Assumption 4. Indeed, comparing the time necessary for direct matching to that required for BK-based matching (i.e., the whole process) shows that exploiting ontologies as background knowledge has a significant computation-time cost, especially when using large BK ontologies as MeSH, or a large number of BK ontologies (see Figures 5.12 and 5.13). However, as demonstrated in the previous chapter (Section 4.6.2), comparing to the traditional approach, our approach reduces the BK selection and anchoring time up to 60%.

5.4 Limitations

As we have mentioned in the previous chapter (Section 4.2.3), to improve our approach efficiency, we consider the smallest ontology as the source ontology. We tried to check whether exchanging the ontology positions (i.e., source ontology becomes target ontology and vice versa) has an impact on the results in terms of Precision, Recall and F-measure.

Theoretically, this may happen when we reuse the BK selection alignments as anchoring alignments (i.e., **Case 1** in Section 4.3). Indeed, the source ontology is anchored to the built BK using the syntactic and structural content of the preselected

ontologies, while the target ontology is anchored to the built BK using only the syntactic information of the selected concepts (i.e., labels). In Figure 5.14, we illustrate this case with an example.

Let O_1, O_2 be two ontologies to align; O_p a preselected ontology; e_1, e_p and e_2 three concepts belonging to O_1, O_p and O_2, respectively. We suppose that matching e_1 to e_p requires structural techniques that exploit the hierarchy of O_1 and O_p, while e_2 can be matched to e_p only with syntactic or lexical techniques.

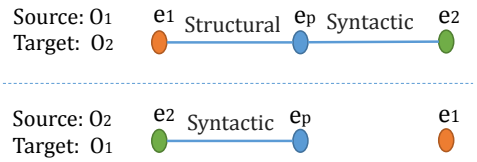

Figure 5.14: Example of exchanging source and target ontologies.

When O_2 is the source ontology, the mapping between e_2 and e_1 cannot be derived, since the structural information of the ontology O_p is not kept in the built BK.

We performed the same experiments as those presented in Section 5.3.3 exchanging source and target ontologies. The results did not change. This may be explained by the fact that the case described above is rare, at least in the used benchmarks. Indeed, discovering mappings based on the structural information is difficult as different ontologies can have different models of the same concept (Pesquita et al., 2013). Usually, the structural information is mainly used to endorse mappings found thanks to syntactic or lexical techniques (Ngo and Bellahsene, 2016, Dragisic et al., 2017).

The effectiveness of the built BK to improve the direct matching alignment depends on two factors: (i) the initial set of preselected ontologies. Indeed, if the preselected ontologies are not semantically rich (e.g., concepts with no synonyms or definitions), and there is no overlap between them and the ontologies to align, our approach, as all the BK-based matching approaches, will not improve the direct matching alignment; and (ii) the quality of the alignments used to build and exploit the BK. Hence, the matcher that generated these alignments (YAM ++ in our case). We suppose that the more effective is the matcher, the higher quality will have the alignments generated by our approach. We plan to study this hypothesis in the future.

We have evaluated our approach on two OAEI benchmarks. Our choice was motivated by the fact that only for these tracks, state-of-the-art systems use ontologies as background knowledge. Hence, evaluating on these benchmarks allows us to compare our results to the state-of-the-art ones. However, these benchmarks include ontologies of one domain, the biomedical domain, which may be considered as a limitation of our evaluation. Nevertheless, the biomedical domain is suitable for evaluating the BK-based matching approaches for two reasons: (i) the vocabulary

of ontologies is complex and specialized, which limits the effectiveness of syntactic similarity measures and generic lexical resources such as WordNet (Faria et al., 2014); (ii) there are many biomedical ontologies with overlapping fragments, which can be exploited as background knowledge.

5.5 Conclusion

In this chapter, we presented the different steps of our BK exploitation approach: (i) anchoring, (ii) derivation and (iii) final mapping selection.

Using BK in ontology matching generates more correct and incorrect candidate mappings. To effectively select the final mappings, we proposed two methods: a rule-based one and an ML-based one. For the second method, we designed a set of 27 attributes to enable the use of an ML classification algorithm.

To evaluate our approach, we have conducted extensive experiments with two OAEI tracks in which BK-based ontology matching systems participate: Anatomy and LargeBio. The obtained results show that:

- The BK built with our approach is effective;
- Our mapping selection methods are effective, and yield almost the same F-measure values. However, ML-based selection promotes precision, while rule-based selection promotes recall;
- The results of our approach are competitive comparing to the state-of-the-art results.

Chapter

6 GBM: Generic BK-Based Ontology Matcher

Contents

6.1 Introduction

Evaluating the benefit of exploiting a given BK can be done only at the end of the matching process by comparing the alignments obtained with and without this BK. Therefore, to perform experiments, one has to deal with the whole BK-based matching process, even if he wants to focus on a specific step such as BK selection or derivation. Indirect matching modules that are implemented in existing matchers such as AML or LogMapBio are tightly related to their internal architectures. Hence, reusing these modules requires a study and an adaptation of their code, which is not always easy. The single generic BK-based matcher was Scarlet (Sabou et al., 2008), however, according to the corresponding author – Marta Sabou –, the Scarlet code is heavily outdated and no more functional.

Taking into account these constraints, we designed a Generic BK-based ontology Matcher (GBM). GBM implements our BK selection and exploitation methods described in Chapters 4 and 5. In addition, we have enriched GBM with new modules to improve the derivation efficiency (see Section 6.4) and generate mappings with relations other than equivalence (see Section 6.3). GBM provides a set of parameters that enables different configurations, and may be easily coupled to any existing matcher. This is particularly interesting to perform experiments.

We have participated in OAEI 2017 and OAEI 2017.5 campaigns with a system, called YAM-BIO, which is GBM with YAM++ as a direct matcher. YAM-BIO obtained good results and was top ranked in several tasks. Moreover, we performed experiments with other direct matchers (LogMap and LogMapLite), to show that GBM – our BK selection and exploitation methods – is generic, and its effectiveness is independent of the direct matcher used.

In the following, we will give an overview of GBM and explain its various parameters in Section 6.2. Then, we will present the method dealing with BK building with internal exploration of the preselected ontologies in Section 6.3. We will describe and evaluate the new derivation algorithm in Section 6.4. After that, we will present and discuss GBM results with YAM++ in OAEI 2017 and OAEI 2017.5 in Section 6.5, and with LogMap and LogMapLite in Section 6.6. Finally, we conclude this chapter in Section 6.7.

6.2 GBM overview

Figure 6.1 shows the five main modules that composes GBM: (i) BK building, (ii) Anchoring, (iii) Derivation, (iv) mapping selection, and finally (v) Semantic verification. We grouped the input parameters in categories (e.g., derivation parameters, selection parameters, etc.). In the following, we will briefly describe the different modules and parameters.

Direct matcher

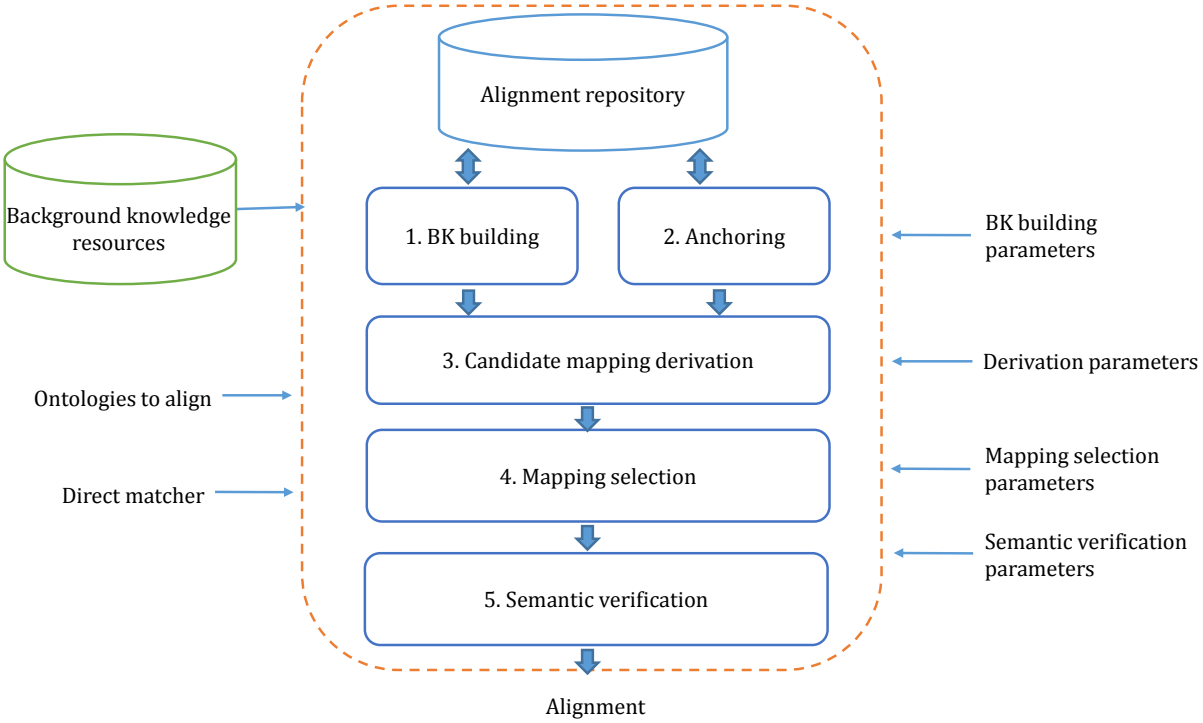

Figure 6.1: GBM architecture.

Any existing direct matcher that implements a basic function *Align*, which takes as input two ontology URLs and returns the URL of the alignment generated by this matcher. The alignment should be stored in RDF format with the API alignment (David et al., 2011) to be parsed correctly. Systems that have participated in the OAEI campaigns, may use GBM directly without any adaptation. Indeed, OAEI participants have to wrap their tools as SEALS packages, and the wrapping procedure includes the implementation of the function Align.[1]

Background knowledge resources

GBM can exploit two background knowledge resource types: (i) ontologies, and (ii) existing mappings. Since we use Jena API to load and parse ontologies, all ontology formats supported by Jena API, such as RDF and OWL, are supported by GBM. However, the formats of the background knowledge ontologies should be supported by the direct matcher too. Existing mappings may be provided in two formats: (a) RDF format: alignments stored using the alignment API or (b) CSV format where each row has a value for the different attributes illustrated in Figure 6.2.

Alignment repository

[1] `http://oaei.ontologymatching.org/2017/`

Attribute	Value
URI source	http://bioontology.org/projects/ontologies/fma/fmaOwlDlComponent_2_0#Abdominal_aorta
Ontology source	http://bioontology.org/projects/ontologies/fma/fmaOwlDlComponent_2_0
URI target	http://purl.obolibrary.org/obo/UBERON_0001516
Ontology target	http://purl.obolibrary.org/obo/uberon.owl
Score	0.99
Relation	=
Source	YAM++

Figure 6.2: Example of an existing mapping format.

As we can see in Figure 6.1, GBM has its own Alignment repository. Indeed, the idea is to avoid aligning the same pair of ontologies with the same matcher more than once to gain in efficiency. Hence, before performing any matching task, GBM verifies if an alignment between the input ontologies exists to reuse it. Otherwise, GBM generates the required alignment and stores it in the alignment repository.

Alignment repository is a folder which contains RDF files, where each file stores an alignment between two ontologies.

BK building The BK building module is the implementation of the approach described in Chapter 4 which takes the two parameters: Direct matcher and Source ontology. In addition, we added a new method that allows to enrich the built BK with internal relations extracted from the preselected ontologies (see Section 6.3).

Derivation GBM provides two mapping derivation strategies. The first one assumes that the Built BK is stored as a Neo4j graph database. It consists in searching all possible paths between source and target concepts. This derivation strategy is complete, i.e., it finds all possible candidate mappings, however it is not scalable for large built BK graphs and it depends on Neo4j. We tried to address these issues by implementing Algorithm 2, which represents the second derivation strategy. In Section 6.4, we explain in detail and evaluate Algorithm 2.

In both cases, the user has to specify the *Maximum path length* parameter, which is the maximum length of paths to be returned by the derivation process (by default it is 4). The length of a given path is the number of its edges.

Table 6.1: Mapping derivation parameters.

Parameter	Possible values
Derivation strategy	All paths or Algorithm 2
Maximum path length	An integer, by default 4

Mapping selection

GBM implements the two mapping selection methods that we have presented in Section 5.2.3: ML based selection and Rule based selection methods. When choosing the ML based selection, the user has to provide one or several datasets, such that each dataset is a folder that contains two ontologies and their validated alignment.

These datasets will be used for training the classifier. When using the second option, the user may specify a threshold value to select only the mappings that have a score equivalent to or higher than this threshold value.

Table 6.2: Mapping selection parameters

Parameter	Possible values
Mapping selection strategy	ML based or Rule based
Threshold	a real value between 0 and 1
Datasets	a folder for each dataset

Semantic verification

Currently, GBM reuses the LogMapRepair module (Jiménez-Ruiz et al., 2013) to verify the consistency of the generated alignment. LogMapRepair takes as parameter the reasoner to use which may be Hermit or Alcomo. The semantic verification is optional and the user may disable it using the *Semantic verification* parameter.

Table 6.3: Semantic verification parameters

Parameter	Possible values
Semantic verification	Yes or No
Reasoner	Hermit or Alcomo

6.3 BK building with internal exploration

Our BK selection approach described in Section 4.2 selects only concepts from the preselected ontologies. Most ontology matching systems generate only equivalence mappings. Hence, using these matchers, the built BK can be exploited to derive only equivalence mappings too. To enable deriving mappings with other relations than equivalence such as subClassOf, we have to enrich the built BK with this kind of relations. To that end, we may extend our BK selection approach to explore the structure of the preselected ontologies and extract fragments, rather than only concepts. The structure exploration is controlled by two parameters:

- the exploration relations. They are the mapping relations that the user wants to generate in addition of equivalence such as subClassOf, and partOf relations.
- the exploration length. This parameter limits the internal exploration within a given preselected ontology to a number of steps. For instance, an exploration with the relation subClassof and length of 1 returns for each concept that has a mapping in the set of filtered mappings (see Section 4.2.3) its parents and children (see Figure 6.3 (b)). If we change the length parameter to 2,

the structure exploration returns for each concept its parents, grandparents, children, grandchildren (see Figure 6.3 (c)).

These parameters may be compared to those proposed in (Locoro et al., 2014) for the local inference step. However, in their work, the authors proposed to reload each BK ontology to explore its structure in the BK exploitation step, which is time consuming. Here, we propose to extract the potentially effective fragments from the preselected ontologies in the BK selection step to prevent dealing or reasoning with complete ontologies in the BK exploitation step.

The result of the structure exploration is a set of triples $\langle e_i, e_j, r \rangle$, such that e_i and e_j belong to the same BK ontology, and r belongs to the exploration relations. These triples are merged with the concepts of the built BK. In the following, we use the term **enriched BK** to refer to a BK built with internal exploration.

Note that a large *exploration length* parameter may return large fragments or whole BK ontologies, which limits the benefit of our BK selection approach. Indeed, our approach aims at extracting the effective BK ontology fragments – as small as possible – rather than returning whole BK ontologies.

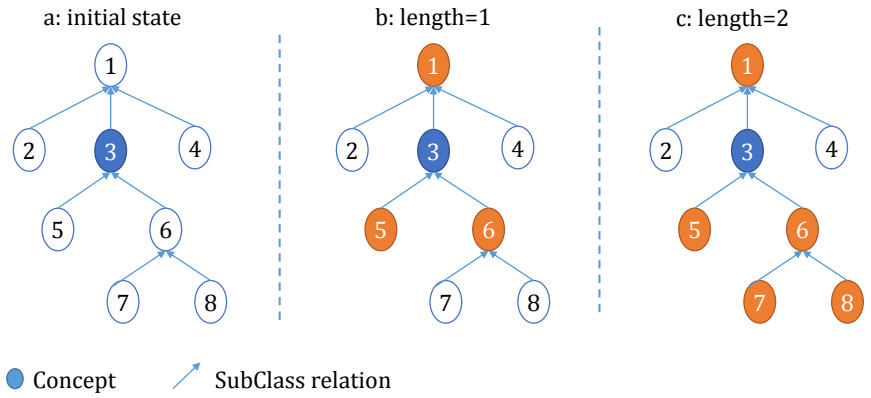

Figure 6.3: Example: concept selection with structure exploration.

In Algorithm 1, we present the pseudo code of the BK building with internal exploration method. Note that this method takes as input a list of source concepts which may be all the source concepts. However, we think it is more efficient to consider only source concepts for which it was not possible to find an equivalent target concept when exploiting the BK built without internal exploration. Indeed, building an enriched BK for the whole source ontology may generate a large graph, which decreases the derivation efficiency, and the precision of the generated alignment.

The implementation of the method explained above offers three new parameters summarized in Table 6.4. When the user assigns *yes* to the parameter *Internal BK ontology exploration*, GBM builds a BK *enriched* with BK ontology relations among the *Internal relations to explore* parameter and the concepts related to these relations within a distance – number of edges – less than the *Exploration length* parameter.

Algorithm 1 Built BK enrichment

Require: $explorationRelations, explorationLength, preselectedOntologies, sourceConcepts$
  $builtBK \leftarrow BKbuilding(sourceConcepts)$
  $enrichedBK \leftarrow builtBK$
  $newConcepts \leftarrow getBKconcepts(builtBK)$
  for all $pOntology \in preselectedOntologies$ **do**
    for all $relation \in explorationRelations$ **do**
      for $i \leftarrow 0; i < explorationLength; i++$ **do**
        $triples \leftarrow getTriples(pOntology, relation, newConcepts)$
        $oldConcepts \leftarrow oldConcepts \cup newConcepts$
        $enrichedBK \leftarrow enrichedBKuniontriples$
        $newConcepts \leftarrow getBKconcepts(triples) - oldConcepts$
      end for
    end for
  end for
  return $enrichedBK$

Preliminary evaluation

To the best of our knowledge, there is no biomedical benchmark to evaluate a matcher tool that returns no-equivalent mappings. Evaluating the BK building with internal-relation enrichment requires the production of a such benchmark by experts. However, due to the lack of time, we performed only a preliminary evaluation on the Anatomy track. We chose Anatomy because it is the smallest and the single biomedical OAEI track that has a manually validated gold standard.

We executed the matching process in two phases: (i) GBM starts by looking for the equivalence mappings following the process illustrated in Figure 6.1 until the derivation step; (ii) then, it builds an enriched BK only for the source concepts that have no equivalent target-concept candidates. We used YAM++ as a direct matcher; we assigned the value 1 to the internal *exploration length* parameter, and rdfs:subClassOf as the *relation to explore* parameter.[2]

Table 6.4: BK Building parameters

Parameter	Possible values
Internal BK ontology exploration	Yes or No
Internal relations to explore	a hierarchical relation e.g., rdfs:subClassOf
Exploration length	an integer

Among the 2744 mouse ontology concepts, 1269 concepts (46%) had no equivalent target-concept candidates. The derivation using the enriched BK returned 965

[2]We used DOID and UBERON as preselected ontologies in all the experiments presented in this chapter.

mappings with subClassOf relation. 517 from the 1269 no-mapped source concepts (41%) have been mapped to target concepts.

We observed that some derived mappings could be inferred from the equivalence mappings without exploiting the enriched BK. For instance, in the mouse anatomy ontology, the concept (MA_0002028;pudendal artery) is a subclass of the concept (MA_0000064;artery). The concept (MA_0000064;artery) has an equivalent concept in the NCIT ontology (NCI_C12372;artery), while the concept *pudendal artery* did not. Thus, it is possible to derive that (MA_0002028;pudendal artery) is subClassOf (NCI_C12372;artery) without exploiting the enriched BK. In the future, we plan to implement techniques to select only the concepts for which no mapping can be inferred.

We manually evaluated 40 subClassOf mappings selected randomly. All the evaluated mappings were correct. We present some examples in Table 6.5. The list of the generated mappings as well as the 40 mappings validated are available in the file *SubClassOfMappings.xls* on GitHub `https://goo.gl/gmGJey`.

Table 6.5: Example of mappings with subClassOf relation between mouse and NCI ontologies.

concept code	preffered label	concept code	preffered label
MA_0000061	arterial blood vessel	NCI_C12679	Blood Vessel
MA_0001871	right atrium valve	NCI_C12729	Cardiac Valve
MA_0000111	annulus fibrosus	NCI_C32599	Fibrocartilage
MA_0000554	thoracic cavity blood vessel	NCI_C12679	Blood Vessel
NCI_C53161	Hyoglossus Muscle	MA_0002296	extrinsic tongue muscle
NCI_C53174	Pronator Teres Muscle	MA_0000615	forelimb muscle
NCI_C53180	Transversus Thoracis	MA_0000548	chest muscle
NCI_C53180	Transversus Thoracis	MA_0000561	thorax muscle

6.4 Candidate mapping derivation algorithm

To participate in the OAEI 2017.5 campaign, we had to execute our algorithms on the Hobbit platform (see Section 6.5). This platform offers limited memory and computational time resources for each execution. Searching all possible paths between source and target concepts – i.e., the first derivation strategy – in a cyclic graph is a complex task which requires significant resources, especially when the graph has a large size. In Algorithm 2, we attempted to reduce the complexity of the all path algorithm by reducing the number of the returned paths. The main idea is to exploit each BK concept once for a given source concept. In Figure 6.4, we present an example to illustrate the difference between the two derivation algorithms. We suppose that all the mappings m_i are equivalence mappings. The all-paths algorithm (case a) returns four paths for the same candidate mapping between the

source concept C_{s1} and the target concept C_{t1}, while Algorithm 2 (case b) returns only one path between the two concepts. Indeed, since C_{BK1} and C_{BK2} have been already exploited to derive the first path, they cannot be reused to derive other paths.

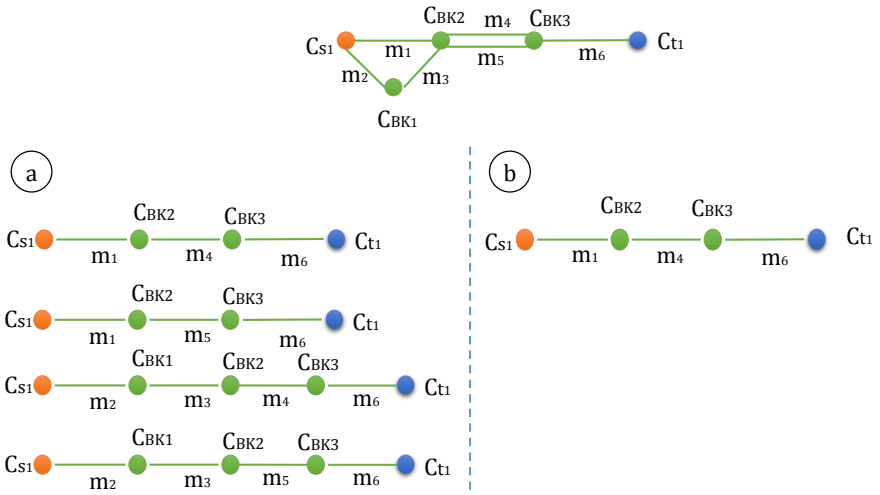

Figure 6.4: Derivation algorithm: all paths vs. Algorithm 2 .

Algorithm 2 promotes short paths, considers only one edge between two concepts. It starts by adding all the filtered mappings and the anchoring mappings to a collection that we call *builtBK*. In this collection, each concept points all the concepts to which it has been mapped directly. After that, for each source concept, we initiate the list of the paths to explore with one path that contains the source concept itself. Then, while it remains paths to explore, we pick up the top path p, we retrieve its last node; if this node has not been *exploited* before, we get the concepts mapped to this node from the *built BK*, we extend p with each of the mapped concepts found to obtain new paths, we verify for each new path whether its last node belongs to the target ontology: if it does, we add the new path to the list of the found paths, otherwise we add it to the list of the paths to be explored if its size is less than the *maxPathLength* parameter.

Evaluation

Figures 6.5 and 6.6 show the results of the two derivation strategies on Task 1 and Task 2 of the OAEI LargeBio track; particularly they show the number of paths, correct and incorrect candidate mappings resulted from the derivation step. In addition, we computed a ratio by dividing Algorithm 1's values by the All-paths values to compare the two derivation algorithms.

As we can see, comparing to the All-paths derivation strategy, Algorithm 1 generates (i) much less paths (37%), (ii) almost the same number of correct candidate mappings (99%), and (iii) slightly less incorrect candidate mappings (82% and 93% in Task 1 and Task 2, respectively). We obtained similar results for the rest of LargeBio tasks and Anatomy.

Algorithm 2 Derivation function

Require: $filteredMappings, sourceConcepts, targetConcepts, maxPathLength$
  Variables:$exploitedConcepts, pathsToExplore, builtBK$
  for all $mapping \in filteredMappings$ **do**
    $addMapping(mapping, builtBK)$
  end for
  for all $sc \in sourceConcepts$ **do**
    $newPath.add(sc)$
    $pathsToExplore.add(newPath)$
    while $pathsToExplore.size() > 0$ **do**
      $path \leftarrow pathsToExplore.get(0)$ # get the top path
      $pathsToExplore.remove(0)$# delete the top path
      $concept \leftarrow path.getLastNode()$
      if $concept \notin exploitedConcepts$ **then**
        $exploitedConcepts.add(concept)$
        $mappedConcepts \leftarrow builtBK.get(concept)$
        for all $mc \in mappedConcepts$ **do**
          $newPath \leftarrow path.add(mc)$
          if $rc \in targetConcepts$ **then**
            $foundPaths.add(newPath)$
          else
            if $newPath.length() < maxPathLength$ **then**
              $pathsToExplore.add(newPath)$
            end if
          end if
        end for
      end if
    end while
    $pathsToExplore.clear()$
    $exploitedConcepts.clear()$
  end for
  return $foundPaths$

The difference in the number of correct and incorrect mappings is explained by the fact that Algorithm 2 does not return paths that includes target ontology concepts as intermediate concepts, while the all path derivation using Neo4j does.

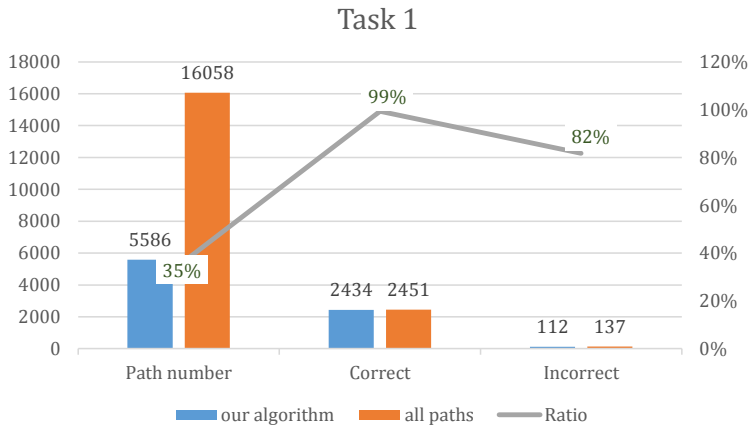

Figure 6.5: Task 1: derivation strategy comparison

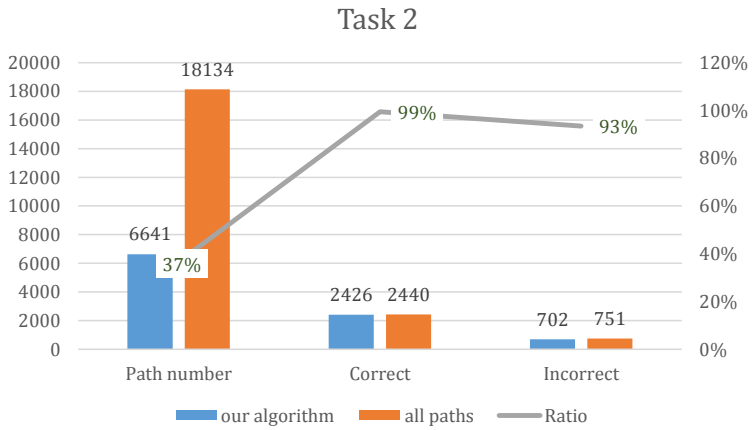

Figure 6.6: Task 2 : derivation strategy comparison

6.5 YAM-BIO results in OAEI 2017 and OAEI 2017.5

We participated in the OAEI 2017 and OAEI 2017.5 campaigns in Anatomy and LargeBio with YAM-BIO as a system, which is GBM with YAM++ as a direct matcher.

In OAEI 2017, we used a basic version of GBM: we used only a set of existing mappings (i.e., OBO mappings) extracted from the UBERON and DOID ontologies as BK, and we applied the indirect matching technique (i.e., mapping composition) only for the source concepts that have not been matched directly (see Figure 6.7).

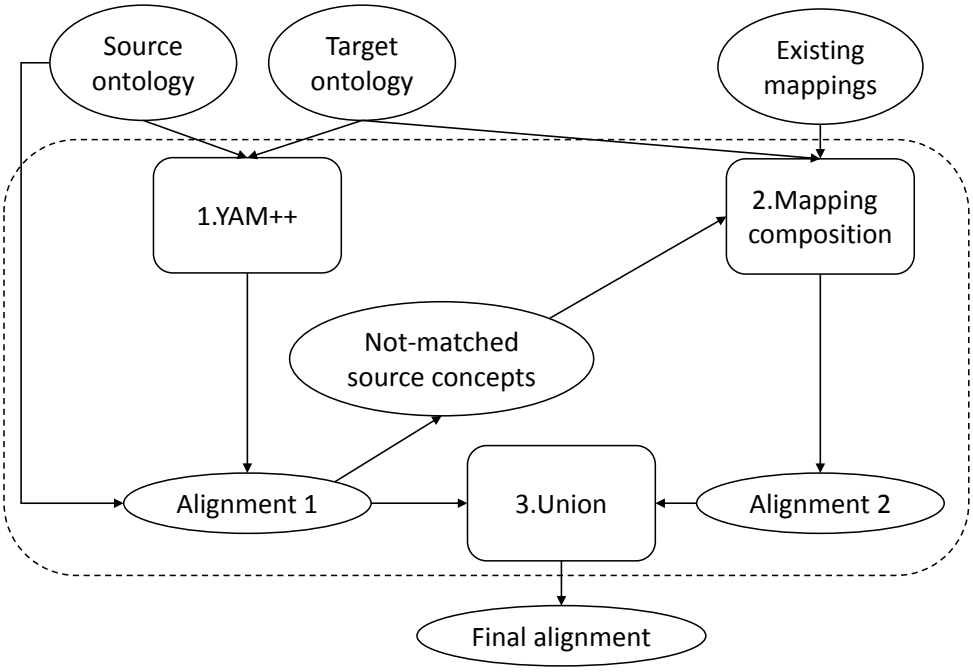

Figure 6.7: OAEI 2017: YAM-BIO architecture.

Exploiting existing mappings as BK allowed YAM-BIO to be scored in second position among the 12 systems that have participated in Anatomy track with almost the same precision and a slightly lower recall than the top ranked system (see Figure 6.8 [3]).[4]

In LargeBio, YAM-BIO was the **top-ranked** system in Task 1 and Task 4 and among the three top-ranked systems in the other tasks. Indeed, it is the second top-ranked system when computing the average of LargeBio results as we can see in Table 6.6. Detailed results are available on the OAEI web page `https://goo.gl/kU5cnK`.

Table 6.6: Average LargeBio results in OAEI 2017.

Matcher	Precision	Recall	F-measure
AML	0.896	0.774	0.827
YAM-BIO	**0.894**	**0.770**	**0.824**
XMAP	0.887	0.760	0.811
LogMap	0.900	0.721	0.797
LogMapBio	0.871	0.734	0.794
LogMapLite	0.858	0.532	0.610
Tool1	0.869	0.367	0.454

[3] `http://oaei.ontologymatching.org/2017/results/anatomy/index.html`
[4] See Section 5.3.3.1 for the original results of YAM++.

Matcher	Runtime	Size	Precision	F-Measure	Recall	Recall+	Coherent
AML	47	1493	0.95	0.943	0.936	0.832	+
YAM-BIO	70	1474	0.948	0.935	0.922	0.794	-
POMap	808	1492	0.94	0.933	0.925	0.824	-
LogMapBio	820	1534	0.889	0.894	0.899	0.733	+
XMap	37	1412	0.926	0.893	0.863	0.639	+
LogMap	22	1397	0.918	0.88	0.846	0.593	+
KEPLER	234	1173	0.958	0.836	0.741	0.316	-
LogMapLite	19	1148	0.962	0.829	0.728	0.29	-
SANOM	295	1304	0.895	0.828	0.77	0.419	-
WikiV2	2204	1260	0.883	0.802	0.734	0.356	-
StringEquiv	-	946	0.997	0.766	0.622	0.000	-
ALIN	836	516	0.996	0.506	0.339	0.0	+

Figure 6.8: Anatomy results in OAEI 2017.

In OAEI 2017.5, we participated with the last version of GBM, described in Section 6.2, on the HOBBIT platform[5] with UBERON and DOID as preselected ontologies.

The Hobbit platform is a generic, modular and distributed platform for Big Linked Data systems. It was designed with the aim of providing an open-source, extensible, FAIR and scalable evaluation platform. However, the platform is not stable yet, and we encountered several technical problems to wrap and run YAM-BIO on it.

In Table 6.7, we report the results that we obtained in the OAEI 2017 (Annane et al., 2017) and OAEI 2017.5[6] campaigns.

Table 6.7: YAM-BIO results in OAEI 2017 and OAEI 2017.5 campaigns.

Campaign	OAEI 2017				OAEI 2017.5			
Task	Precision	Recall	F-measure	T(s)	Precision	Recall	F-measure	T(s)
Anatomy	0.948	0.922	**0.935**	70	0.946	0.913	0.929	176
Task 1	0.968	0.896	0.931	56	0.971	0.902	**0.935**	197
Task 2	0.816	0.888	0.852	279	0.818	0.894	**0.855**	518
Task 3	0.966	0.733	0.834	60	0.962	0.741	**0.837**	244
Task 4	0.887	0.728	0.800	468	0.879	0.738	**0.802**	755
Task 5	0.899	0.677	0.772	220	0.927	0.703	**0.800**	478
Task 6	0.827	0.698	0.757	490	0.842	0.697	**0.763**	962

Generally, OAEI 2017.5 results are better than those obtained in OAEI 2017. The precision improvement may be explained by the use of the rule based selection

[5] https://master.project-hobbit.eu/home

[6] Results may be consulted on the HOBBIT platform https://goo.gl/A496ug

method and the semantic verification. Indeed, in OAEI 2017, we kept all the mappings generated indirectly in the final alignment, and we did not use any semantic verification technique what generated a high incoherence degree in some tasks such as Task 3. However, these mapping selection techniques affected Anatomy recall and F-measure. Such a negative impact of mapping selection techniques on the final results has already been reported in the literature. For instance, when the input ontologies have different modeling views, the semantic verification my eliminate correct mappings (Pesquita et al., 2013).

In OAEI 2017, we did not generate any alignment other than the one between the ontologies to align, we simply composed the existing mappings. However, in OAEI 2017.5, we generated the different alignments required for the BK building and the BK exploitation steps, which improved the recall – especially in Task 3 and 5 – and increased the computation time. Note that the computation time is not directly comparable since the experiments have been performed on two different platforms SEAL and HOBBIT. However, it is trivial that YAM-BIO in OAEI 2017.5 consumed more computation time because it performed more matching tasks.

According to YAM-BIO results in OAEI 2017 and OAEI 2017.5, we may suppose that considering only the concepts that have not been matched directly in the BK-based matching, may be a **tradeoff** – or a compromise – between the matching quality (i.e., F-measure) and the computation time, which decreases slightly the F-measure scores, and allows to gain in efficiency. This hypothesis should be confirmed with deeper experiments.

OAEI 2017.5 campaign was aiming to test the Hobbit platform by the matching systems. Hence, most of participants such as AML or LogMap have reused the OAEI 2017 versions. Additionally, because of the technical constraints imposed by this evaluation platform such as the maximum computation time, some systems have not participated such as LogMapBio since it requires much time to select background knowledge ontologies from NCBO BioPortal. Therefore, we compare our OAEI 2017.5 results to the participant results in OAEI 2017.

With an F-measure of 0.929, YAM-BIO is the **third top-ranked** system in Anatomy (see Figure 6.8), and with an average F-measure of 0.832, YAM-BIO is the **top-ranked** system in LargeBio (see Tables 6.6 and 6.8).

Finally, we may note that there no ideal matching strategy, and it depends on the user needs in terms of precision, recall and computation time. For instance, even if YAM-BIO has almost the same F-measure (0.763) as AML in Task 6, AML has a higher precision 0.904 vs. 0.842, while YAM-BIO has a higher recall 0.697 vs. 0.668.

Table 6.8: Average LargeBio results in OAEI 2017.5.

Measure	Precision	Recall	F-measure
YAM-BIO	0.900	0.779	0.832

Comments on the OAEI evaluation

When possible, we think it would be interesting to publish participants results with and without exploiting of specialized background knowledge resources. On one hand, this will allow to better evaluate the influence of background knowledge on matching results and computation time. On the other hand, this will allow a fair comparison with systems that do not use background knowledge.

Some components are common in all ontology matching system architectures; others do not always exist – such as background knowledge resource selection or semantic verification. This makes the comparison of computation time particularly cumbersome and not always fair. According to us, it would be more appropriate to evaluate execution times for each separate component. For example, YAM-BIO used a predefined background knowledge while LogMapBio made a dynamic selection from an online repository necessarily taking additional time. Splitting running time by components will also help the community to identify less efficient components to improve them, and most efficient ones to reuse them.

6.6 GBM with LogMap and LogMapLite matchers

To verify the effectiveness of GBM with other matchers than YAM++, we performed the same experiments replacing YAM++ by LogMap and LogMapLite matchers. LogMap applies consistency principles and LogMapLite essentially applies direct string matching techniques. We report the obtained results in Tables 6.9 and 6.10.

Comparing to the original results of LogMap, GBM shows slightly better results (F-measure) in almost all tasks, except in Task 2 because of a low precision. This low precision may be explained by the use of the preselected ontology UBERON (see Section 5.3.2 for more detail). Note that LogMap exploits UMLS lexicon, a rich biomedical lexicon, as external knowledge resource, which reduces the benefit of using other external biomedical knowledge resources. Indeed, the improvement is more significant with LogMapLite, especially in Tasks 3 and 4 (see Table 6.10).

Table 6.9: LogMap: original results vs. results with GBM.

TASK	Original results				Results with our framework			
Measure	Precision	Recall	F-measure	T (s)	Precision	Recall	F-measure	T (s)
Anatomy	0.918	0.846	0.880	4	0.900	0.947	**0.923**	42
Task 1	0.944	0.897	0.920	7	0.945	0.896	**0.920**	53
Task 2	0.856	0.808	**0.831**	53	0.763	0.851	0.804	174
Task 3	0.947	0.690	0.798	43	0.924	0.735	**0.819**	104
Task 4	0.840	0.645	0.730	302	0.798	0.695	**0.743**	465
Task 5	0.947	0.69	0.798	192	0.924	0.705	**0.800**	332
Task 6	0.868	0.597	0.707	622	0.795	0.683	**0.735**	923

Figure 6.9 shows two series: (1) diff1: the difference between the LogMap F-measure values and the LogMapLite F-measure values; (2) diff2: the difference

Table 6.10: LogMapLite: original results vs. results with GBM.

TASK	Original results				Results with our framework			
Measure	Precision	Recall	F-measure	T (s)	Precision	Recall	F-measure	T (s)
Anatomy	0.962	0.728	0.829	1	0.929	0.921	**0.925**	24
Task 1	0.967	0.819	0.887	1	0.963	0.860	**0.909**	25
Task 2	0.673	0.820	0.739	7	0.674	0.841	**0.748**	59
Task 3	0.968	0.209	0.344	2	0.942	0.394	**0.555**	29
Task 4	0.852	0.209	0.336	12	0.822	0.393	**0.532**	92
Task 5	0.892	0.567	0.693	6	0.924	0.667	**0.774**	62
Task 6	0.797	0.567	0.663	12	0.818	0.658	**0.730**	116

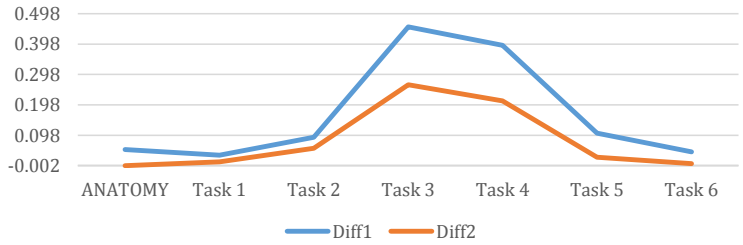

Figure 6.9: F-measure difference: original results vs. our framework results.

between our framework results when using LogMap and LogMapLite. As we can see, in all matching tasks diff1 is less than diff2, which means that exploiting the BK allowed to reduce the gap between the two matchers (a sophisticated matcher and a simple matcher).

6.7 Conclusion

In this chapter, we presented a generic BK-based Ontology Matcher, called GBM, which implements the main contributions of our thesis. GBM offers several parameters and may be easily customized according to the user needs. GBM is publicly available to the community in a GitHub project.[7]

We have used YAM-BIO – GBM with YAM++ – matcher to participate in the OAEI 2017 and OAEI 2017.5 campaigns in two tracks: Anatomy and LargeBio. The results obtained in those tracks were very close to top-ranked state-of-the-art systems, thanks to the different content matching techniques implemented in YAM++, the BK used and our BK selection and exploitation methods. Furthermore, we performed experiments with two other matchers LogMap, and LogMapLite. In both cases, GBM improved the original alignments generated by these matchers. This demonstrates that the effectiveness of GBM is independent of the direct matcher used.

[7] `https://github.com/AminaANNANE/GenericBKbasedMatcher`

Currently, the set of preselected ontologies is provided manually. In the future, it will be interesting to integrate a new module that preselects automatically a set of background knowledge ontologies. To that end, we plan to evaluate the performance of an existing ontology recommender such as the NCBO ontology recommender (Romero et al., 2017).

General Conclusion and Open Issues

Chapter

7 Conclusion and Open Issues

Contents

This chapter aims to summarize the main contributions of this thesis and to outline a number of directions for future work. We start with Section 7.1 to highlight our contributions to the list of research questions initiated in the introduction of this thesis. Next, in Section 7.2, we present some open issues related to the topic of our thesis.

7.1 Main contributions

In this thesis, we focused on enhancing direct ontology matching by exploiting external knowledge resources (BK). Our contributions concerned the two main issues related to BK-based matching: (i) the selection of an effective BK for a given ontology matching task and (ii) the exploitation of the selected BK in the matching process. In the following, we summarize our contributions.

7.1.1 Review of BK selection and BK exploitation methods

In the literature, several works have dealt jointly or separately with the BK-based ontology matching issues. Hence, it was crucial to analyze and compare the existing works. Studying these works allowed us to define a common BK-based ontology matching workflow in Chapter 2. The workflow includes two main components related to the two main issues (i.e., BK selection and BK exploitation) and, takes as input the two ontologies to align and a set of knowledge resources that we called *knowledge resource pool*. While the BK selection process is different from one work to another, the BK exploitation is common and composed of three sub-tasks which are: anchoring, candidate mapping derivation and, mapping aggregation and selection. We used this workflow to describe and compare the existing works in Chapter 3. Initially, BK selection is performed manually, however the quantity and size of the available knowledge resources motivated the community to develop automatic BK selection methods. This automatic selection varies from a simple search on the web to more sophisticated methods based on similarity measures. In BK exploitation, the variety in methods is mainly observed in the mapping derivation process. To facilitate the comparison, we elaborated a classification based on three criteria: (i) the structure exploration, (ii) the number and, (iii) combination of the exploited resources. In addition, we tried to evaluate the impact of using external knowledge resource in ontology matching using the results of the 2012-2016 OAEI campaigns. Unfortunately, systems exploiting BK participate only in the biomedical tracks which limits the evaluation to be domain independent. However, since we are mainly interested in matching biomedical ontologies, we estimate that this evaluation is reliable in the context of our thesis. The study showed that systems exploiting BK obtain better results than systems that do not, especially by yielding a higher recall. In addition, we observed that indirect matching techniques lead to lower precision and longer computation time.

7.1.2 A novel efficient and effective BK selection/building method

As we pointed out in chapter 3, ontologies are the most appropriate external knowledge resources. In addition, in the biomedical domain, there is an abundance of ontologies with overlapping fragments. Therefore, we chose to use a set of biomedical ontologies as a knowledge resource pool.

Existing automatic BK selection methods return complete ontologies as background knowledge to be used in the BK exploitation step. Our hypothesis was that, for a given matching task, only fragments from the effective BK ontologies are actually effective , and returning complete ontologies, especially large ones, decreases the efficiency of the BK exploitation process. In this thesis (see Chapter 4), we proposed to build dynamically one knowledge resource by selecting and combining fragments from the preselected ontologies (i.e., ontologies of the knowledge resource pool). BK building is based on the reuse of alignments (i) between the source ontology and the preselected ontologies, and (ii) the preselected ontologies between each other. These alignments may be generated by an automatic matcher or extracted from an existing resource such as UMLS. The built BK depends only on the source ontology. Consequently, for the same knowledge resource pool, the same built BK may be reused to match this source ontology to any target one. Moreover, the built BK enables the derivation across one or several intermediate BK ontologies. We conducted experiments on two well known OAEI benchmarks, with two sets of preselected ontologies, which demonstrated (i) the reduced size of the built BK, and (ii) the gain in efficiency obtained thanks to our BK selection approach.

7.1.3 BK exploitation methods

The built BK has a graph format which allows to use a graph database to derive candidate mappings. Indeed, to search candidate mappings, we proposed to search all paths between source and target concepts. This solution is complete since it returns all the possible paths which may be interesting for the final mapping selection methods, however, it is not scalable for large graphs. Therefore, we implemented a derivation algorithm that reduces the number of derived paths up to 37% (see Chapter 6).

Furthermore, in Chapter 5, we proposed two methods to select the final mappings from the candidate ones. A rule-based and ML-based methods. For the second method, we created a set of 27 attributes that describe each candidate mapping to allow the use of a classification ML algorithm. The two methods provide almost the same results in terms of F-measure. However, according to the experiments, the ML-based selection promotes precision while rule-based selection promotes recall. The rule-based mapping selection is simple and efficient but static. Indeed, the same rules are applied all the time although each ontology matching task has its own specificities. For instance, the best threshold value varies from one matching task to

another. The ML-based mapping selection is time consuming and requires aligned ontologies to generate the training data. However, it allows to build automatically a customized classifier that combines many selection attributes (27 in our case). Our experimental results confirm the effectiveness of our BK exploitation approach, which improved significantly the direct matching results and overcomes or competes with state-of-the-art matchers exploiting background knowledge resources.

7.1.4 GBM: Generic BK-based Matcher

Finally, the output of this thesis is a generic BK-based ontology matching framework, which is the implementation of our BK selection and BK exploitation methods (see Chapter 6). This framework is configurable and may be easily integrated to any existing matcher. GBM participated to the OAEI 2017 and OAEI 2017.5 in biomedical tracks where it was top ranked in several matching tasks.

7.2 Open issues

We have identified several research directions that deserve a deeper study.

7.2.1 Generating semantic mappings

One of the advantages of BK-based ontology matching is its ability to generate semantic mappings i.e., mappings with semantic relationships such as *disjoint* and *subClassOf* (Sabou et al., 2008). This kind of mappings allows reasoning with alignments, which is one of the ontology matching challenges (Shvaiko and Euzenat, 2008). We recently implemented the BK-building with internal exploration module (see Chapter 6). Preliminary results are promising; however, more in-depth experiments should be conducted to evaluate the performance of the generated mappings and study the impact of some parameter variations, such as the internal exploration length parameter, on the generated results. To realize this evaluation, we plan to work with experts to develop a gold standard including mappings with different relationships. The produced gold standard could be reused to introduce a new OAEI matching task.

7.2.2 Extending the evaluation to other domains

In this thesis, we evaluated our BK-based ontology matching approach on seven biomedical matching tasks (see Chapter 5); we obtained promising results, however, to demonstrate that our approach is domain-independent, we have to perform more experiments on benchmarks of other domains than biomedical. One possible target domain is agriculture. Indeed, in the context of the AgroPortal project (Jonquet et al., 2018), a repository that groups ontologies related to agriculture has been created. As observed in the biomedical domain, it exists many mappings stored

in several resources such as AGROVOC (Caracciolo et al., 2013). In the future, we plan to evaluate our approach in matching agriculture ontologies and possibly implement it as a service on the AgroPortal repository.[1]

7.2.3 User interaction

A graphical user interface (GUI) is necessary to help the user verify and validate the mappings generated automatically. Moreover, we believe it would be interesting for the user to visualize the part of the built BK graph that allowed deriving a given mapping. This could provide explanations about how this mapping has been derived. Indeed, explaining the generated mappings is recognized as an ontology matching challenge (Shvaiko and Euzenat, 2008). Furthermore, such a GUI will enable the framework usage by biomedical researchers that are not necessarily computer scientists. The validated mappings may be stored and reused as a BK for further matching tasks. We are aware that manual validation of all the mappings generated is fastidious, and may require a long time, especially for large matching tasks such as Task 6 of the OAEI LargeBio track (17210 mappings in the reference alignment). One challenging issue is to minimize the expert intervention such that by validating a small set of candidate mappings the matcher can automatically validate or not the remaining candidate mappings. Recently, machine learning community tends to propose methods using weak supervision strategies (Bach et al., 2017, Ratner et al., 2017). The idea, called data programming, consists in generating automatically large training data from a small one. In the future, it would be interesting to investigate data programming in ontology matching.

7.2.4 Exploiting other resource types than ontologies

Most BK-based ontology matching methods exploit ontologies as background knowledge (see Chapter 3). However, there are not as many ontologies in all domains as in the biomedical domain. Hence, it would be interesting to exploit other knowledge resource type, particularly textual resources, which are available in all domains without exception. Textual resources are not structured which makes their exploitation difficult. The recent advances in natural language processing, especially in named entity recognition and relation extraction may facilitate the exploitation of textual resources by offering methods and tools allowing to construct structured knowledge resource from text (Niu et al., 2012, Song et al., 2015).

7.2.5 Derivation scalability

In our approach, deriving candidate mappings consists in finding paths between source and target nodes (i.e., concepts) through the built BK graph. Depending on the size of the ontologies to align, the number, the size and the overlapping of the

[1] `http://agroportal.lirmm.fr/`

preselected ontologies, the built BK may become a large cyclic graph. In such a case, deriving all possible candidate mappings requires more memory and time. Scalability of algorithms dealing with processing and analyzing large graphs such as social networks, protein networks, and LinkedData graphs, is a known research issue and one of the most timely problems facing the big data research community (Batarfi et al., 2015). A common approach, evaluated recently to improve SPARQL queries, is to ask not for all, but only for the k shortest paths (Savenkov et al., 2017). In the future, it would be interesting to evaluate the performance of such algorithms in terms of matching quality (i.e., F-measure) and computation time using a large knowledge resource pool such as NCBO BioPortal (more than 700 biomedical ontologies).

7.2.6 Combining several matchers

Currently, several matchers are available such as YAM++, AML and LogMap. However, according to the OAEI results, not all matchers find the same correct mappings. In addition, none of them is able to achieve good results across all matching tasks. Hence, we estimate that it would be more effective to combine alignments generated by different matchers in our BK-based ontology matching approach. However, such a combination, raises several research issues such as the selection of the matchers to be used, their number, and the combination of the mappings generated by the selected matchers. These issues have been already highlighted by the ontology matching community in the context of combining several similarity measures within the same matcher (Shvaiko and Euzenat, 2008), and several methods have been proposed (Duchateau and Bellahsene, 2016). It would be interesting to review the proposed methods and, if necessary improve them or develop new ones, to evaluate the impact of combining several matchers on the alignments produced by our approach.

Appendices

Appendix

A. Multilingual Mapping Reconciliation

Contents

A.1 Introduction

The biomedical domain is rich in terms of ontology[1]. However, the majority of these ontologies are in English (Névéol et al., 2014) and even when ontologies are available in other languages like French, there is a strong lack of related tools and services to use them. This lack does not reflect the huge amount of biomedical data produced, especially in the clinical world (e.g., electronic health records). The repository of biomedical ontologies NCBO BioPortal (http://bioportal.bioontology.org) (Noy et al., 2009) includes, as of end 2015, more than 433 ontologies, only six are not in English, five in French and one in Spanish (Jonquet et al., 2015). Furthermore, the UMLS (Unified Medical Language System) Metathesaurus (Bodenreider, 2004), even if it covers 21 languages, 75.1% of its terms are in English and only 1.82% of its terms are in French (Bollegala et al., 2015). There have been initiatives in the past to reinforce the involvement of French language in the UMLS (Darmoni et al., 2003, Zweigenbaum et al., 2003) but most of these French ontologies are still not included, they are most often aggregated and translated by the CISMeF group[2] (Grosjean et al., 2011) (324.000 French concepts in HeTOP vs. 85,000 in the native UMLS)[3]. The lack of support for ontologies in different languages represents a real barrier for non-English-speaking communities that produce and manage biomedical data in their own languages. Indeed, when biomedical resources contain text content, it is important that these resources' languages are the same as the language of the ontologies that will help to index or exploit them. Hence there is the need to have multilingual or translated ontologies (Meilicke et al., 2012, Fu et al., 2009, Deléger et al., 2009). The translation of MeSH by the French organization INSERM[4] is a good example and has greatly enriched the French biomedical vocabulary in UMLS (Névéol et al., 2014). However, except in Meta-thesaurus approaches such as the UMLS or CISMeF where ontologies are integrated in a common model, when someone gets a translated ontology to work with, it is never formally aligned to the original one and there is no standard format or resource to get such alignments. It definitively prevents multilingual use of ontologies for annotation, semantic search, and data indexing neither for integration or knowledge extraction from these data. To ensure semantic interoperability, it is not enough to just translate ontologies, we must also formally keep the link between objects of the translated ontologies and the original ones (Buitelaar et al., 2009). Re-establishing this link is the aim of this work, which we have called reconciliation of multilingual mappings. These multilingual mappings, once established and represented in a formal way, can have multiple applications (Fu et al., 2010). For example, they allow performing a multilingual indexing of biomedical resources, which allow multilingual semantic

[1] In this work we use ontology to identify both of the (biomedical) terminological and ontological resources.

[2] Rouen's University Hospital (http://www.chu-rouen.fr/cismef/)

[3] (http://www.hetop.eu/hetop/)

[4] http://www.inserm.fr/

search. A user types in a query using French terms and retrieves results within English data resources (and vice-versa). Multilingual mappings also allow integrating biomedical data of different languages. For example, resistance to diseases differs from one population to another, and it is a research problem that could be studied at a larger scale thanks to the multilingual mappings which enable cross-language databases integration. Indeed, the correlation study between genotypes and diseases (Köhler and al., 2014) across different populations databases, annotated each in its original language with biomedical ontologies, linked by multilingual mappings, allow researchers to have a better vision of the problem and potentially, to discover new knowledge.

Our work is part of the SIFR project (Semantic Indexing of French Biomedical Data Resources - http://www.lirmm.fr/sifr) in which we are interested in exploiting ontologies in construction of services like indexing, mining, and information retrieval for French biomedical resources. In this project, we develop a semantic indexing workflow (called the French Annotator) based on ontologies similar to that existing for English resources (Jonquet et al., 2009), but focused on the French resources. To improve the workflow and connect the used French ontologies to their English equivalents, the project focuses on the reconciliation of multilingual mappings.

The present study concerns ten French ontologies hosted on the SIFR BioPortal (http://bioportal.lirmm.fr) (a local instance of BioPortal dedicated to French) that we wish to align formally with their original English ontologies hosted on the NCBO BioPortal. The idea is to be able to retrieve from a French concept in the SIFR BioPortal, its corresponding English concept in the NCBO BioPortal and vice versa. As of now we are mainly focusing on *monolingual* ontologies but in parallel we are studying how to manage multilingualism in BioPortal (Jonquet et al., 2015).

The rest of the chapter is organized as follows: Section 2 is dedicated to the presentation of related work in the field. Section 3 describes our approach to represent multilingual mappings within the BioPortal architecture. Section 4 presents the ontologies used in our work. Section 5 explains the followed methodology. Then, section 6 exposes obtained results. Section 7 discusses the study and its results. Finally, section 8 concludes and presents the perspectives of this work.

A.2 Related work

Multilingualism has always been considered as an important issue for the semantic web (Buitelaar and Cimiano, 2014), that has even became more important with the explosion of data. Several challenges are identified (Gracia et al., 2012), in particular cross-lingual ontology alignment and the representation of multilingual lexical information in ontologies, which are the starting points for the cross-lingual access and querying of linked data. In the following, we briefly review related work on these two issues. In the literature (dos Santos et al., 2014), several approaches have been proposed to extract multilingual mappings. The first was the manual

approach where mappings are extracted by human experts as in the work of Liang and Sini (Liang and Sini, 2006), who manually aligned the English version of the AGROVOC thesaurus to the Chinese Agriculture Thesaurus. Despite the accuracy of the mappings generated by this approach, it cannot be used to process large and complex ontologies. Therefore, researchers have turned to the automated approaches using different techniques: machine learning (Spohr et al., 2011b), machine translation (Fu et al., 2012), extraction mappings using multilingual background (Tigrine et al., 2015), etc. Overall, the ontology alignment community mostly focuses on the topic of generating mappings between different ontologies in different languages (dos Santos et al., 2014,Euzenat and Shvaiko, 2013) and ignores the problem of mapping reconciliation considered (truly) as a more easy issue. However, the reality shows us that: (i) it is not that trivial: ontologies and their translation are always different (they do not follow the same evolution after the process of translation) and (ii) the community still needs those mappings out there for use in concrete applications.

On the other hand, there have been several attempts to define models representing the linguistic description of terminological resources on the web (thesaurus, ontologies, etc.). The RDFS model allows to represent labels of concepts through the rdfs:label property without more information. SKOS model refine this property and decompose it into three properties which are *preferred label*, *alternative label* and *hidden label*. However, these properties are not enough to describe the linguistic characteristics and in particular cross-lingual specifications. To fill these gaps, other models were proposed such as: the GOLD ontology (General Ontology for Linguistic Description) (Farrar and Langendoen, 2003), which allows to represent formal linguistic concepts using an OWL ontology. The Lemon model (LExicon Model for ONtologies) model (McCrae et al., 2011a), which is now the most widespread representation for the publication of lexical resources as linked data. Indeed, Lemon is the result of the evolution of several models: LMF (Lexical markup framework) model, LexInfo model (Cimiano et al., 2011) and Linguistic Information Repository (Montiel-Ponsoda et al., 2008). Lemon allows describing more information on lexica, in particular: morphology, phrasing structure and subcategorization information. It also allows representing lexical information relative to an ontology that is shared on the semantic web. It has been gradually expanded to include new modules such as translations (Gracia et al., 2014) resulting in the newly developed model OntoLex/Lemon (Bosque-Gil et al., 2015). It is really good to have such models to represent all linguistic details of lexica, but we also need to think about the use of proposed models. Rich models such as Lemon are complex to implement. Indeed, details as parts of speech, morphology, etc. need linguistic experts to determine them and formalize them. This task is very hard, especially for large and complex ontologies like SNOMED-CT. Consequently, there is a need to specific tools to support these models use in order to convince stakeholders in the web of data to adopt them (Gracia et al., 2012).

As of now, the biomedical domain is one of plenty of ontologies that are not being lexically grounded, and are not multilingual. For which a translation has sometimes

been produced by another group/project than the group that has developed the original one (e.g., MeSH, MedlinePlus, ICD, MEDDRA, and ICPC). Many of these ontologies are made available within the NCBO BioPortal (Noy et al., 2009) but this platform is not multilingual even if it accepts both multilingual and monolingual ontologies (Jonquet and Musen, 2014). Another important resource in the biomedical domain, the UMLS Metathesaurus, which is a set of terminologies manually integrated and distributed (mostly publicly) by the United States National Library of Medicine (Bodenreider, 2004). It does contain terminologies in other languages than English and therefore, explicitly store the mappings between them. However, the number of French resources in the UMLS is not sufficient to cover the diversity of the biomedical domain. The HeTOP portal (Grosjean et al., 2011) also offers translated terms in multiple languages, especially French, and enables cross lingual search but most of its content is not publicly or easily accessible (e.g., No web service API or ontology download functionality). Furthermore, in both cases, the underlying approach is one of a common meta-model for all the integrated ontologies which means that there exists a unique abstraction for concepts in different sources (e.g., the UMLS Concept Unique Identifiers (CUI)) and label properties offer translations to multiple languages. This is different from the BioPortal approach that we are also following. This approach does not build a global thesaurus but keep each ontology separated and use mappings to interconnect them (Noy et al., 2008, Ghazvinian et al., 2009a). Another difference with BioPortal, is that neither UMLS nor HeTOP are built natively with semantic Web technologies and thus do not offer semantic representation to make multilingual ontologies or multilingual mappings available as linked data. The review of the state of the art identifies (i) the need (at least for French) for an explicit reconciliation of the multilingual mappings between translated ontologies and their origin ones and (ii) the need for making them available as linked data.

A.3 Multilingual mappings in BioPortal

Our aim is to link the French ontologies hosted on SIFR BioPortal with their English counterparts hosted on the NCBO BioPortal. For this purpose, we need to represent multilingual mappings (Jonquet et al., 2015) in a way that will ensure the interoperability between the two portals and avoid duplicating the data.

A.3.1 Choice of the mapping properties

BioPortal stores mappings in a particular format that reifies a mapping as a RDF resource. These mappings can have several properties including provenance information (process, note, date, who created, etc.). Especially, BioPortal uses one property of standard semantic web vocabularies to tag / describe a mapping between two concepts of ontologies . For example, the property skos:exactMatch to indicate that

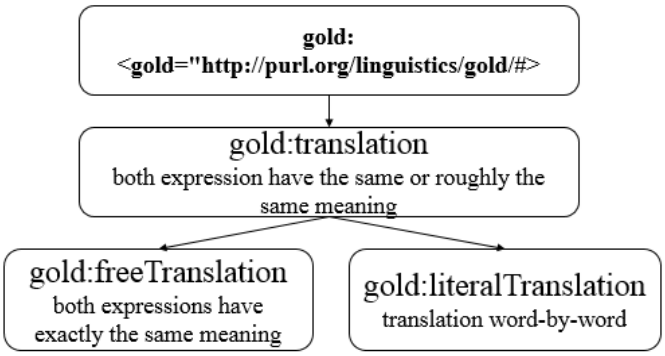

Figure A.1: Translation properties of GOLD ontology

two concepts are identical and the property skos:closeMatch to mark a strong bond of similarity between two concepts without being completely identical. With the mapping being reified, the triple e.g., (Cls1 skos:exactMatch Cls2) is not explicitly materialized but can be retrieved any time. Indeed, all mappings (as any other data in the portal) are available either via a SPARQL endpoint or via JSON-LD. We propose to represent multilingual mappings as any mapping in the portal, but with specific additional semantic properties to mark the linguistic aspect and formalize the translation relationship between two concepts. For example, the concept Mélanome (http://purl.lirmm.fr/ontology/MSHFRE/D008545) in the French version of MeSH within the SIFR BioPortal should be mapped to concept Melanoma in the English version of MeSH (http://purl.bioontology.org/ontology/MESH/D008545) in the NCBO BioPortal. This allows to specify that the two concepts Mélanome and Melanoma have the same meaning and that the first label is the French translation of the second one. We can still use the SKOS properties to represent that the two concepts have the same meaning. Nevertheless, for the linguistic information, we need another property to describe the translation relationship. For this purpose, we propose to use Lemon or GOLD models. In the following, we have chosen to use the GOLD properties, in particular gold:freeTranslation property to represent an accurate translation, and gold:translation to represent a less precise translation (see Figure A.1). As of now, we did not use Lemon as no other lexical or linguistic layer was necessary for our biomedical ontologies.

A.3.2 Changes in BioPortal architecture

In order to store our multilingual mappings, we had to change their representation in the BioPortal architecture, especially: (1) Allow to tag the same mapping with several semantic web properties to avoid duplicating the mappings (semantic mapping and translation mapping); (2) Allow a BioPortal virtual appliance to store mappings that target ontologies (i) in another instance of BioPortal (inter-portal), (ii) that are not in any BioPortal instance (external mappings). In order to for-

mally figure out the source and target languages of a translation, we assumed the metadata of the monolingual ontologies would mention the natural language used for labels. Indeed, BioPortal offers the property omv:naturalLanguage included in the OMV ontology (http://omv2.sourceforge.net) which uses ISO-639-3 to specify the appropriate language for each ontology.

A.4 Ontologies to align

We have treated a set of 20 ontologies, 10 in French and 10 in English. These ontologies are widely used in the biomedical field both in French and in English. For example, the International Classification of Diseases (CIM10ICD10) is used in hospitals to code medical acts, the Medical Subject Heading (MeSH) is used for indexing documents both by the NLM (English) and CISMeF (French). In our study, all English ontologies come from the UMLS Metathesaurus (version 2015AA) and were imported by the NCBO team in the NCBO BioPortal using the umls2rdf tool (https://github.com/ncbo/umls2rdf). The French ontologies come from the UMLS or were provided by the CISMeF group as an OWL file. In this second case, the translations were generally produced or synthesized by CISMeF. All processed ontologies are stored in the SIFR BioPortal (see Table A.1).

A.5 Methodology

The followed methodology consists of: (1) Download ontology files in .ttl or .owl formats from the NCBO and SIFR BioPortals. (2) Parse them with the Jena API to extract the necessary data for multilingual alignment. (3) Store the data in SQL table (one table per ontology). (4) Make the relevant *join* queries between the two tables on the field/property used to reconcile the mappings. (5) Finally, post the produced mappings to SIFR BioPortal after choosing the relevant GOLD and SKOS properties (see Figure A.2).

A.5.1 Downloading files

For this study, we have chosen ten ontologies for which we have a French version in the SIFR BioPortal and that contains labels that will be easily used by the SIFR Annotator for identifying biomedical words in text. These ontologies have been downloaded from English and French BioPortals. As an example, files of the English and the French version of the SNOMED International terminology (SNMI) are respectively available at: https://bioportal.bioontology.org/ontologies/SNMI, http://bioportal.lirmm.fr/ontologies/SNMIFRE.

Table A.1: Ontologies processed in this study (acronyms are identifiers from the NCBO BioPortal and the SIFR BioPortal)

N°	Ontology	Acronym	Version	Source
1	Systematized Nomenclature of MEDicine	SNMI	2015AA	UMLS
	Systematized Nomenclature of MEDicine, version française	SNMIFRE	3.5	CISMeF
2	International Classification of Functioning, Disability and Health	ICF	1.0.2	UMLS
	Classification Internationale du Fonctionnement, du handicap et de la santé	CIF	2001	CISMef
3	MedlinePlus Health Topics	MEDLINEPLUS (EN)	2015AA	UMLS
	MEDLINEPLUS FR	MEDLINEPLUS (FR)	-	CISMeF
4	Minimal Standard Terminology of Digestive Endoscopy	MSTDE	2015AA	UMLS
	Terminologie minimale standardisée en endoscopie digestive	MTHMSTFRE	2011ab	UMLS
5	Semantic Types Ontology	STY (EN)	2015AA	UMLS
	Réseau sémantique UMLS	STY (FR)	2014AB	CISMeF
6	Medical Subject Headings	MESH	2015AA	UMLS
	Medical Subject Headings, version francaise	MSHFRE	2015AA	UMLS
7	Medical Dictionary for Regulatory Activities	MEDDRA	2015AA	UMLS
	Dictionnaire médical pour les activités règlementaires en matière de médicaments	MDRFRE	2015AA	UMLS
8	World Health Organization (WHO) Adverse Reaction Terminology	WHO	2015AA	UMLS
	World Health Organization (WHO) Adverse Reaction Terminology, version française	WHO-ARTFRE	1997	CISMeF
9	International Classification of Diseases, Version 10	ICD10	2015AA	UMLS
	Classification Internationale des Maladies, version 10	CIM-10	10	CISMeF
10	International Classification of Primary Care - 2 PLUS	ICPC2P	2015AA	UMLS
	Classification Internationale de Soins Primaires	CISP-2	1998	CISMeF

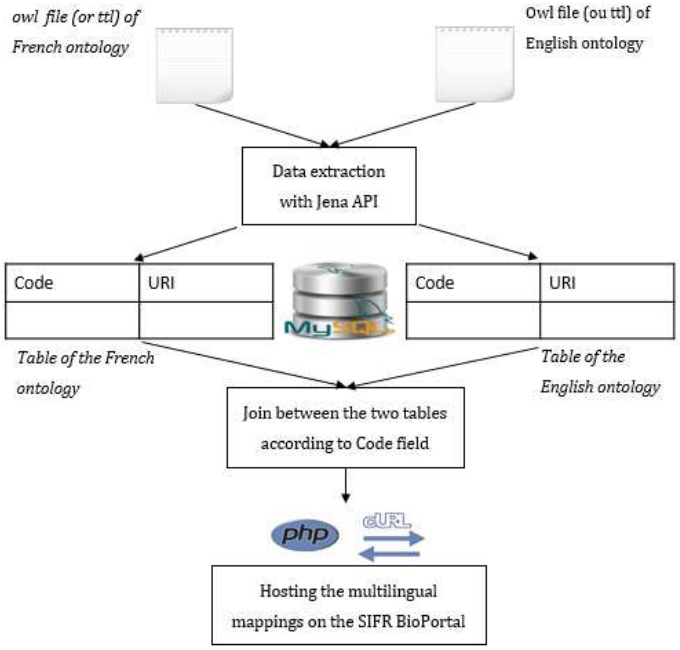

Figure A.2: Overview of the multilingual mapping reconciliation process

A.5.2 Retrieving data from ontologies files

We used the Jena API (https://jena.apache.org) to extract RDF triplets (concept, property, propertyValue) from the ontology files. To extract only the needed triplets, we filtered them according to a specific property. Mostly, this property is the field that we are going to use to reconcile mappings. Indeed, ontologies always provide a unique identifier or code for the concepts / classes they define. With recent OWL ontologies, this is of course an URI, but this was not the case for ontologies that have been originally designed not following a semantic web principles. To determine the appropriate property, we had to study the ontologies one by one. In most cases, the alignment was based on the internal code of concepts in ontologies. Except MedlinePlus, as there is no internal code for its concepts, we had to use the UMLS code (CUI). However, the code property name differs from one ontology to another, the most frequent was skos:notation, used for 12 of the 20 ontologies. Other specific properties have been used such as icpc2p:icpccode for ICPC2P or icd:icdCode for ICF. In the five cases where a code property did not exist, we extracted the code from the URI of classes (those were always provided as the files were extracted from BioPortal). In these cases, in order to retrieve only classes corresponding to a concept in the ontology, we filtered and retained only the classes that have a property skos:prefLabel. Technically, we have developed a script (function) to extract the code used for each alignment. Eventually, more treatments were necessary such as elimination of the type attached to the value or use of regular expression to isolate the exact code string. Table 2 summarizes the property used to retrieve the mapping

code with examples for each ontology.

A.5.3 Saving data

To store the data extracted in the previous step, we have created a table within a relational database for each ontology. Each triplet retrieved adds a new record in the appropriate table. For example, the triplet (http://purl.bioontology.org/ontology/MSHFRE/D001542; skos:notation; D001542) extracted from the French version of MeSH generates the record shown in Table 4 with the three columns:

- Id: a sequential number that identifies each record in the table
- Code: a string that contains the code previously extracted and which may be an internal code of the concepts in the ontology, the CUI or any other relevant mapping criteria.
- URI: globally identifies the concept. This URI is either assigned by the ontology designers or created by the BioPortals.

There is no unique constraint on the relational table for the two fields *Code* and *URI*. This is justified by the fact that for a given URI it is possible to have multiple codes, and for a given code we can have multiple URI. Even if this situation should not occur in theory, it actually does happen in practice. Especially with properties such as CUI that are not specific to the ontology but had been added when the ontology was integrated with other ones in the UMLS Metathesaurus. For instance, this is the case with the French version of the MEDLINEPLUS ontology, which contains 442 concepts that have more than one CUI. For example the concept minéraux (http://chu-rouen.fr/cismef/MedlinePlus#T4298) has nine different CUI. Consequently we have nine triplets that have the same concept (same URI) but different codes (different CUI) which involves the creation of nine records that have all the same URI. We also encountered cases where the code targets multiple URIs within the same ontology (ICPC2P, MedlinePlus FR, and CIM10). We will address these cases in more details in the Results section.

A.5.4 Reconciliation of mappings

At this stage, we do a *join* clause on the *Code* field between the two corresponding tables. As we mentioned in the previous paragraph, the code used in the join is not necessarily unique within the same ontology. Consequently, the number of couples (fr URI, en URI) resulted from the join can be more than the number of concepts of one of the ontologies (or both) (see Figure 3). For example, the CISP2 ontology has generated 5063 mapping couples whereas it has only 745 concepts. The multiple assignment of a code to several resources generates also duplicated couples that we eliminated (using the SQL keyword "distinct" in the join query) because it represents only redundant information (see Figure 4).

A.5.5 Mapping property selection and loading in SIFR BioPortal

This final step allows representing and storing produced mappings in a formal and permanent way within the SIFR BioPortal. The particularity of our multilingual mappings, compared to other BioPortal mappings resides in the couple of properties by which they were tagged (GOLD translation property and SKOS alignment property) as follows:

skos:exactMatch/gold:freeTranslation: We used these properties when the French concept has the same meaning as the English concept. This is usually the case when the mapping is based on full equality between the internal code of the French concept and the internal code of the English concept. In that case, there are no more mappings generated than the minimum number of classes between the two ontologies being mapped. Fortunately, it is the most frequent case.

skos:broadMatch/gold:translation : We used these properties when the source concept is more precise than the target concept. This situation occurs when the translated ontology was modified (e.g., new sub-concepts more precise). For example, the French concept agression par un moyen non précisé établissement collectif (http://chu-rouen.fr/cismef/CIM-10#Y09.1) from CIM-10 which has internal code Y09.1, do not have an English concept having the same internal code in ICD10. However, we can map it to the concept Assault by unspecified means which has internal code Y09 (http://purl.bioontology.org/ontology/ICD10/Y09) and tag this mapping with these properties.

skos:narrowMatch/gold:translation: We used this couple of properties when the target concept is more precise than the source concept. This situation occurs when the original ontology has been modified after the translation into French. We had this case in the alignment of CISP2 to ICPC2P through the icpc-code (internal code). Concepts of ICPC2P have the icpc-code following by another code (term-code), which is not the case for the French ontology CISP2 that only uses original icpc-code. Consequently, one French concept is mapped to several English concepts, which have the same icpc-code. For example, the concept labelled tumeur bénigne having B75 as a code, has been mapped to eight English concepts more precise, three among them are (benign neoplasm of the blood, B75001), (benign neoplasm of the lymphatics, B75002), (benign neoplasm of the spleen, B75003). Hence, the need to use such properties.

skos:closeMatch/gold:translation: In the absence of an internal code, we had to use less precise identifiers such as the CUI for ontologies coming from UMLS. CUIs are identifiers at the Metathesaurus level, and not in the sources ontologies. Therefore, it is not a direct translation of the concept from one language to another but rather concepts that mean the same thing as they were affected to the same CUI. The Table A.2 summarizes the semantic properties used to tag multilingual mappings between two concepts in our study. The first column indicates the type of translation equivalence as it was identified by Chen and Chen [35], the second

one describes possible relationships between concept and the last one indicates the semantic properties used to tag our multilingual mappings. For more information about how multilingual mappings are represented and stored on SIFR BioPortal see section 6.9.

Table A.2: Summary of semantic properties used to describe the multilingual mappings

Translation	Description	Properties
exact	Two equivalent concepts	skos:exactMatch gold:freeTranslation
inexact	Two concepts very similar but not equivalent	skos:closeMatch gold:Translation
partial	Target concept is less precise	skos:broadMatch gold:translation
	Target concept is more precise	skos:narrowMatch gold:translation

A.6 Results

Our aim is to provide multilingual alignment for French versions of ontologies. We express our results as a percentage of the French ontology concepts for which, we were able to provide at least one translation mapping (see Table A.3).

STY/STY, MDRFRE/MEDDRA, CIF/ICF: For these three couples of ontologies, we got a full alignment, one to one for all concepts (percentage of 100%) thanks to the internal code of concepts.

MTHMSTFRE/MSTDE Among the 1700 concepts of the French version, only 2 concepts have not been mapped since their codes do not exist in the English version. These two concepts have as an internal code MT180041 and MT180042. In the English version, there is one concept that has not been mapped. Its internal code is set to *nocode*. However, this concept has two CUI which are those assigned to the unmapped French concepts. Thus, the mapping of these two French concepts was done according to their CUI, which allowed us to get a percentage alignment of 100%. We estimate that this 'nocode' case is an error in the integration of MSTDE in UMLS and we will share this with NLM.

MSHFRE/MESH: The number of concepts in the English version (252242 concept) is ten times greater than the number of concepts in the French version (26142 concept), because the French version contains only the MeSH descriptors without the other additional concepts . Our mappings cover almost all French concepts with a percentage of 99.79%. Only 55 concepts of the French version have not been aligned because their codes do not exist in the English one such as D020185 (Acides benzoïques), D002134 (Protéine de liaison au calcium dépendant

de la vitamine D), D006587 (Acides hexanoïques). Moreover, even trying to align these 55 concepts using CUI, we have not found their CUI in the English MeSH. When CUI are used in MSHFRE but they do not exist in MESH, we think they are probably mistakes that were made by translators or problems which appeared during the integration of the new translation into UMLS. Indeed, the two versions should be perfectly aligned since they both come from UMLS 2015AA. To correct this problem we will communicate the 55 non-mapped concepts to INSERM and NLM.

Table A.3: Summary of results

Fr ontology	Concepts	En ontology	Concepts	Mapped concepts	Mapping %	# Mappings	Properties (skos ; gold)
STY	133	STY	133	133	100%	133	exactMatch; freeTranslation
MDRFRE	66378	MEDDRA	66378	66378	100%	66378	exactMatch; freeTranslation
CIF	1495	ICF	1495	1495	100%	1495	exactMatch; freeTranslation
MTHMSTFRE	1700	MSTDE	1699	1700	100%	1700	exactMatch; freeTranslation
MSHFRE	26142	MeSH	252242	26220	99.79%	26220	exactMatch; freeTranslation
WHO-ARTFRE	3482	WHO	1724	3482	100%	3482	broadMatch; translation
CISP2	745	ICPC2P	7537	665	70%	5063	narrowMatch; translation
MEDLINEPLUS	795	MEDLINEPLUS	2113	771	97%	1520	closeMatch; translation
CIM-10	19853	ICD10	12318	19813	99%	19813	exactMatch; freeTranslation 62% broadMatch; translation 37%
SNMIFRE	106266	SNMI	109150	102093	96%	102093	exactMatch; freeTranslation

WHO-ARTFRE/WHO-ART: In the English version WHO-ART, the internal code of concepts can be retrieved through the skos:notation property. But in the French version, this code cannot be found in any property of WHO-ARTFRE; we had to extract it from URIs of concepts. Indeed, this code is located at the end of the URI of each concept. For example, the code of the concept with URI http://chu-rouen.fr/cismef/WHO-ART#1545_PT is 1545. We have noticed that the French version has undergone some customization. Indeed, a code of the English version can reference several French sub-concepts that have the same code but suffixed to differentiate them. For example, the code 1723 references four concepts that have the following codes: 1723_IT0, 1723_IT1, 1723_IT2, 1723_PT. Therefore, the number of French concepts is greater than the number of English concepts (3320 vs 1724). The French version is more detailed, their concepts are more precise than those of the English version, so we used the two properties (skos:broadMatch ; gold:translation) to describe mappings between these ontologies. Finally, all the French concepts were mapped.

CISP2/ICPC2P The French version CISP2 contains 745 concepts while the number of generated mappings was 5141. This is explained by the fact that the English version which has been modified in this time after the French translation. English concepts have been customized to generate new more detailed concepts (sub-concepts of the original ones: 7354 concepts). Therefore, an icpc-code of the French version was mapped with one or several English concepts that have the same icpc-code but differentiated through another code called "term code". For example, the code A01 is affected to a single concept in the French version (Douleur générale/de sites multiples; A01) while the English version contains four more precise concepts as follows: (generalised aches; A01001), (generalised pain; A01003), (body pain; A01004), (chronic pain; A01005). Consequently, a single concept of CISP2 may generate several couples of mapping, one for each English concept that has the same icpc-code. For this reason, we used the properties (skos:narrowMatch; gold:translation) to describe the mappings. 59 of the French ontology concepts have not been aligned with the icpc-code such as: (Autre analyse de laboratoire; 38), (Conseil thérap/écoute/psychothérapie; 58), (Examen microbiologique/immunologique; 33). We figured out these concepts do not have an icpc-code as the rest of the concepts that consists of a letter followed by two digits. In addition, these concepts have no CUI property as well. It seems they have been added in the translation, or removed from the English version. For this reason we reached only 70% of mappings for CISP2 and we will communicate our results to the translators.

MEDLINEPLUS FR/EN We had to use the CUI property to align MEDLINEPLUS as its concepts have no other internal code . The French version of MedlinePlus contains 795 concepts. Each concept has one or more CUI value (442 concepts have more than one) which gives 1686 distinct couples (concept, CUI). The English version contains 1986 distinct concepts and each concept has a single CUI value. Indeed, the URI of each English concept is suffixed by the CUI assigned to

it, for example C0003803 is the CUI of the URI
http://purl.bioontology.org/ontology/MEDLINEPLUS/C0003803.
It is surprising to note that there are CUIs that do not exist in the English version but are assigned to concepts of the French version. However, even if we ignore the concepts with these CUI (147 concept), the number of couples (concept, CUI) remain greater than the number of concept (1520 couples vs 795 concepts). 123 concepts of these 147 have other CUIs belonging to the English version but the remaining 24 concepts have no CUI belonging to the English version. So these concepts do not exist or no longer in the English version. Therefore, 24 French concepts are not mapped and we obtain a percentage alignment of 90% in terms of aligned couples and 97% in terms of aligned concepts. We have tried to refine the study, for eight among concepts that do not appear in the English version at all, we applied the following procedure:

1. Search the preferred label in the French ontology;
2. Translate manually the term, using the terminology portal TermSciences[5] or another lexical resource (e.g., BabelNet or even simple Google translation), into English;
3. Search, in the English version, the obtained translated term and if the English corresponding concept exists, note its CUI.

As we can see in the Table A.4, in seven cases over eight, we found the English concept, which corresponds to the French concept but with a different CUI. These results make us think that these 24 unmapped concepts are mistakes in the CUI choice during the translation process. We intend to communicate these concepts to the translators in order to detect possible errors and possibly update their translation.

CIM10/ICD10 CIM10 contains 19853 concepts while its English version, ICD10, contains 12318 concepts. Here again, we figured out that the French version has undergone some customizations; it was enriched with more detailed concepts resulting from specialization of the original concepts. A "join" clause according to the internal codes of concepts between the two ontologies generated mapping percentage of 62% (12 308 concepts were mapped). We observed that there are six chapters in the French version, CIM-10, that do not have the same internal code as their English counterparts such as chapter (autres maladies infectieuses; B99) in CIM-10, while in ICD10 the same chapter is (Other infectious diseases; B99-B99). These chapters have the characteristic to contain only one entry. We had to treat them manually since the join according the code field did not work. All of the previous mappings (automatic and manual) were tagged with properties (skos:exactMatch; gold:freeTranslation). As for the concepts generated by specializations (which codes do not exist in the English version), we extracted the code

[5] www.termsciences.fr

Table A.4: Correspondences between unmapped French concepts and English concepts

Fr CUI	Preferred Label	En CUI	Proffered Label
C0156543	Avortement	C0392535	Abortion
C2362506	Fitness et exercice	C1456706	Fitness and Exercise
C0021311	Infections	C3714514	Infections
C1456593	santé mentale et comportement	C1832070	mental health and behavior
C1456620	vivre avec le SIDA	C2963182	Living with HIV/AIDS
C1456571	nutrition des nourrissons et des bébés	not found	"nutrition of infants and babies"
C2362562	sécurité du patient	C1113679	patient safety
C0002808	Anatomie	C0700276	Anatomy

of their direct unique parent concept (the first 3 digits of their internal code) and mapped them with the correspondent English parent concepts tagging them with the properties (skos:broadMatch ; gold:translation). For example, all the French concepts (Agression par d'autres moyens précisés /domicile ; Y08.0), (Agression par d'autres moyens précisés/ établissement collectif ; Y08.1), (Agression par d'autres moyens précisés /lieu de sport et d'athlétisme ; Y08.3) were mapped with the English concept (Assault by other specified means ; Y08). By following this process, we reduced the number of unmapped concepts from 7545 to 40 concepts, which gives 99% of mapped French concepts.

SNMIFRE/SNMI The French version SNMIFRE has 106266 concepts, while the English version contains 109150 concepts; there is a difference of 2884 concepts. Using the internal code, 102093 French concepts have been mapped (96% of the French ontology). However, there remained 4173 concepts of the French version without mapping. We tried then to use the CUI property, but those 4173 concepts are part of a set of 9510 French concepts that do not have this property (whereas all concepts of the English version does have a CUI). We have not found another relevant field to use for mapping the remaining 4173 concepts.

Multilingual mappings hosted on SIFR BioPortal All alignments produced in our study are hosted on the SIFR BioPortal with a script that uses SIFR BioPortal REST web service API[6]. As a result, for all ontologies processed during this work, when browsing a concept (see Figure A.3), we can see in the *Class Mappings* tab the multilingual alignments classified as *Interportal mappings* with a flag to indicate that it is a linguistic mapping to English, we can also observe the properties used. The aligned concept link allows the user to switch from the SIFR BioPortal to the target concept in the NCBO BioPortal. Like all the content of the SIFR BioPortal, in addition to the graphical interface, these multilingual mappings

[6]http://data.bioportal.lirmm.fr/documentation

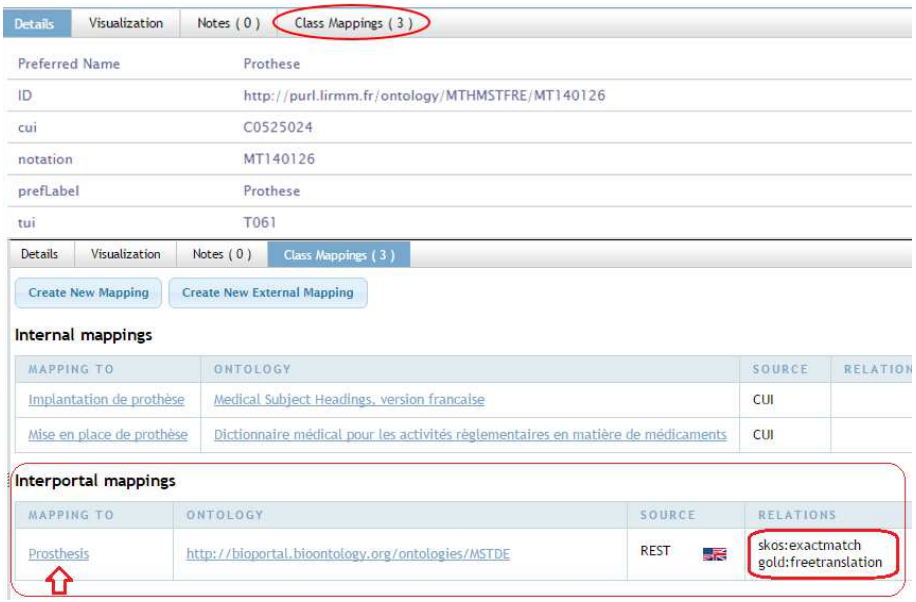

Figure A.3: Example of a multilingual mapping for the concept Prothèse in MTHMSTFRE within the SIFR BioPortal.

are also available directly via the REST web service API and a SPARQL endpoint which makes them part of the web of data; easily readable and reusable by any semantic web applications.

A.7 Discussion

In this work, we propose an approach to formally represent semantic links between translated ontologies and their original ones. Particularly, we focused on French ontologies hosted within the SIFR BioPortal and their English counterparts hosted within the NCBO BioPortal. Our approach consists in reconciling and representing these links as multilingual mappings using semantic web properties. However, this work should not be confused with multilingual mapping extraction that consists in aligning two different ontologies, which have no relationship with each other and which are not in the same language. Indeed, in most of our cases, we have used internal codes to reconcile links. Hence, the semantic link between the translated concept and its origin existed implicitly through the internal code despite difficulties we have met in certain cases. Our mission was to restore these links and represent them in a formal way and publicly made them available where the ontologies actually reside. However, our approach to represent and store the mappings can be used to represent any kind of mappings either reconciled or extracted assuming the relevant semantic properties will be used. In our case, we have chosen SKOS and GOLD properties. They are complementary, especially in the linguistic aspect. Indeed, the gold:translation does not represent the difference between the narrow translation,

broad translation or close translation (see Table A.2), but combining with SKOS properties we have the exact description. For example the couple (skos:narrowMatch ; gold:translation) describes inexact translation of type narrow. We could also have tagged the best mappings with the owl:sameAs property because in theory the concept is exactly the same, and their logical entailment should be equivalent. However, we did not want to take the risk to assign such a property without experimentally verifying that no other inconsistencies will show up. We therefore left it to future users the choice of considering those mappings as owl:sameAs when materializing the triples e.g., (Cls1 tag Cls2) out of BioPortal's mapping repository. It is necessary to evaluate the result of an alignment process (dos Santos et al., 2014,Euzenat and Shvaiko, 2013) to be able to use them. However, since we did a reconciliation of mappings, we have restored links between concepts based on internal code of concepts and not on a terminological, structural or semantic measures (Shvaiko and Euzenat, 2013). Consequently, our approach gives automatically reliable and verifiable results. Indeed, as we can see in Figure A.4 92% of the produced mappings are the result of total equivalence of concepts internal code and 7% of partial equivalence (the internal code of the French concept is included in the internal code English concept or the reverse). We do acknowledge that the remaining 1% of MEDLINEPLUS mappings had to be verified because of the multiple affectation of CUI to a given concept in the French version, which is not the case in the English version (see section 6.4). For example, the concept santé au travail pour les professionnels de santé has two CUIs (C1456673, C0206333), therefore, it was mapped with two English concepts (Blood-Borne Pathogens, C0206333) and (Occupational Health for Healthcare Providers, C1456673).Whereas in this case, only the second target concept is correct. What causes the error was the wrong affectations of CUIs and our work should help the translator of the ontology to fix them. It is important to note that even if the community produces less and less 'monolingual' ontologies and that designers are opting increasingly for "multilingual ontologies"; we cannot assume that ontology translation will not happen anymore. Indeed, regardless of the richness of an ontology in terms of language (2, 3 even 10 languages), it would never cover all languages. Translated ontologies remain then an ineluctable solution to ensure their exploitation in other languages that are not supported in native version. We hope this study will convince ontology translators about the importance of reusing the same identifiers when creating a translated version. Eventually, the best situation is to follow the semantic web principles and actually reuse the exact same URI, when available, rather than creating a new one. Furthermore, in the process of creating multilingual ontologies, there is still a challenge of going further than the simple use of the xmllang tags and move to using lexical standards models such as Lemon.

The multilingual mapping links produced in this study can have several applications including the integration of biomedical data of different languages, and multilingual semantic search and indexing. In the continuation of the SIFR project, these links will be integrated into the French version of the NCBO Annotator (Jon-

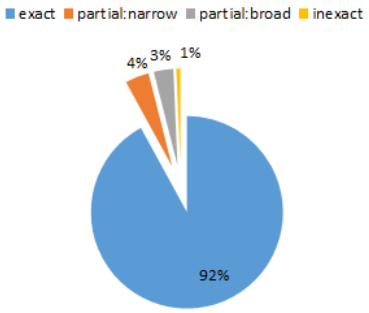

Figure A.4: Distribution of multilingual mappings per type

quet et al., 2009) which will expand direct annotations with French ontologies to new annotations with (i) their corresponding English ontologies, (ii) other English-only ontologies mapped one another inside the NCBO BioPortal. In addition, our mappings will also be a good corpus for automatic translation of biomedical ontologies i.e., they can help translators themselves to translate more ontologies.

Although this is not mostly the case for the current 20 selected ontologies, we have to assume both the original and the translated ones could be modified in the future . Therefore, it is necessary to implement a strategy to keep multilingual mappings up to date (Hartung et al., 2008). Currently, we run the script again when a new version is available; we remove all the old multilingual mappings to store the new ones. This processing will be done automatically once the script of multilingual mapping reconciliation will be integrated directly into the SIFR BioPortal.

A.8 Conclusion

Ontologies play a key role in the semantic interoperability of biomedical data. To reuse them, they have often been created in a particular language(s) and then translated to other language(s). Indeed, the use of ontologies for annotation, search and indexing of data is strongly linked to the syntactic correspondence between ontologies languages and the data languages. Hence, there is a need of multilingual management to allow the exploitation of knowledge formalized in the ontologies in languages other than their original ones.

In this work, we performed alignment reconciliations; we have restored formal semantic links between ten translated French ontologies and their English counterparts using semantic properties of SKOS and GOLD vocabularies. Finally, all these mappings (228k) are stored on the SIFR BioPortal platform (http://bioportal.lirmm.fr/mappings) and they are available to the scientific community as linked open data through a SPARQL endpoint and also as a web service API that returns JSON-LD format. In a near future, we also plan to process LOINC (Logical Observation Identifiers Names and Codes) that was recently made available in French in UMLS. To accomplish this work, we had to treat each pair of ontology apart with its specificities espe-

cially in the choice of alignment property and how to recover it. Refinements were needed when translated ontologies did not follow exactly the content of the original ontology (in English). Through this study, we have found some anomalies in certain pairs of ontologies which we intend to communicate to the translators in order to review them and eventually correct them. The current listing of the anomalies and the concepts that are concerned is available as well as our reconciliation scripts and the data used at: https://github.com/sifrproject/multilingual_mappings. This work represents a part of the SIFR project aiming to efficiently manage multilingualism in a repository of biomedical ontologies such as NCBO BioPortal. As future work, will use these mappings for the development of a process to infer ontologies translations automatically based on multilingual ontologies, different dictionaries and Metathesaurus like UMLS, BabelNet, etc. We will also work on the valorization of these mappings in services such as indexing, annotation and semantic search. Another interesting work to achieve will consist in materializing some of the reified mappings (exact mappings) into owl:sameAs direct mappings and use a reasoner to check possible inconsistencies in the whole repository of interconnected ontologies.

Bibliography

Achichi, M., Cheatham, M., Dragisic, Z., and al. (2016). Results of the ontology alignment evaluation initiative 2016. In *11th International Workshop on Ontology Matching, OM, Kobe, Japan*, pages 73–129.

Aleksovski, Z., Klein, M., Ten Kate, W., and Van Harmelen, F. (2006a). Matching unstructured vocabularies using a background ontology. In *15th International Conference on Knowledge Engineering and Knowledge Management, EKAW, Podebrady, Czech Republic*, pages 182–197.

Aleksovski, Z., Ten Kate, W., and Van Harmelen, F. (2006b). Exploiting the structure of background knowledge used in ontology matching. In *1st International Workshop on Ontology Matching, OM, Athens, Georgia, USA*, pages 13–24.

Amini, R. (2016). *Towards Best Practices for Crowdsourcing Ontology Alignment Benchmarks*. PhD thesis, Wright State University.

Annane, A., Bellahsene, Z., Azouaou, F., and Jonquet, C. (2016a). Selection and combination of heterogeneous mappings to enhance biomedical ontology matching. In *20th International Conference on Knowledge Engineering and Knowledge Management, EKAW, Bologna, Italy*, pages 19–33.

Annane, A., Bellahsene, Z., Azouaou, F., and Jonquet, C. (2017). YAM-BIO: results for OAEI 2017. In *12th International Workshop on Ontology Matching, OM, Vienna, Austria*, pages 201–206.

Annane, A., Bellahsene, Z., Azouaou, F., and Jonquet, C. (2018). Building an effective and efficient background knowledge resource to enhance ontology matching. *Journal of Web Semantics*, 51:51 – 68.

Annane, A., Emonet, V., Azouaou, F., and Jonquet, C. (2016b). Multilingual mapping reconciliation between english-french biomedical ontologies. In *6th International Conference on Web Intelligence, Mining and Semantics, WIMS, Nîmes, France*, pages 13:1–13:12.

Annane, A., Emonet, V., Azouaou, F., and Jonquet, C. (2016c). Réconciliation d'alignements multi-lingues dans bioportal. In *27es Journées francophones d'Ingénierie des Connaissances, IC, Montpellier, France*, pages 23–34.

Bach, S. H., He, B. D., Ratner, A., and Ré, C. (2017). Learning the structure of generative models without labeled data. *CoRR*, abs/1703.00854.

Batarfi, O., Shawi, R. E., Fayoumi, A. G., Nouri, R., Beheshti, S.-M.-R., Barnawi, A., and Sakr, S. (2015). Large scale graph processing systems: survey and an experimental evaluation. *Cluster Computing*, 18(3):1189–1213.

Bellahsene, Z., Emonet, V., Ngo, D., and Todorov, K. (2017). YAM++ online: a multi-task platform for ontology and thesaurus matching. In *14th Extended Semantic Web Conference, ESWC, Posters and Demonstrations, Portoroz, Slovenia*.

Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The semantic web. *Scientific american*, 284(5):28–37.

Bizer, C., Heath, T., and Berners-Lee, T. (2009). Linked data - the story so far. *Semantic Web Journal*, 5(3):1–22.

Bodenreider, O. (2004). The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32:267–270.

Bollegala, D., Kontonatsios, G., and Ananiadou, S. (2015). A cross-lingual similarity measure for detecting biomedical term translations. *PloS one*, 10(6):e0126196:1–28.

Bosque-Gil, J., Gracia, J., Aguado de Cea, G., and Montiel-Ponsoda, E. (2015). Applying the ontolex model to a multilingual terminological resource. In *4th Workshop on the Multilingual Semantic Web (MSW), Portorož, Slovenia*, pages 27–38.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Buitelaar, P. and Cimiano, P., editors (2014). *Towards the Multilingual Semantic Web, Principles, Methods and Applications*. Springer.

Buitelaar, P., Cimiano, P., Haase, P., and Sintek, M. (2009). Towards linguistically grounded ontologies. In *The Semantic Web: Research and Applications, 6th European Semantic Web Conference, ESWC 2009, Heraklion, Crete, Greece, May 31-June 4, 2009, Proceedings*, pages 111–125.

Caracciolo, C., Stellato, A., Morshed, A., Johannsen, G., Rajbhandari, S., Jaques, Y., and Keizer, J. (2013). The AGROVOC linked dataset. *Semantic Web*, 4(3):341–348.

Chauhan, A., Vijayakumar, V., and Ragala, R. (2015). Towards a multi-level upper ontology/foundation ontology framework as background knowledge for ontology matching problem. *Procedia Computer Science*, 50:631–634.

Cheatham, M., Dragisic, Z., Euzenat, J., Faria, D., Ferrara, A., Flouris, G., Fundulaki, I., Granada, R., Ivanova, V., and Jiménez-Ruiz, E. (2015). Results of the ontology alignment evaluation initiative 2015. In *10th International Workshop on Ontology Matching, OM, Bethlehem, USA*, pages 60–115.

Cheatham, M. and Hitzler, P. (2013). String similarity metrics for ontology alignment. In *12th International Semantic Web Conference, ISWC, Sydney, Australia*, pages 294–309.

Chen, S. and Chen, H. (2012). Mapping multilingual lexical semantics for knowledge organization systems. *The Electronic Library*, 30(2):278–294.

Chen, X., Xia, W., Jiménez-Ruiz, E., and Cross, V. (2014). Extending an ontology alignment system with BioPortal: a preliminary analysis. In *13th International Semantic Web Conference, ISWC, Posters and Demonstrations, Riva del Garda, Italy*, pages 313–316.

Cimiano, P., Buitelaar, P., McCrae, J. P., and Sintek, M. (2011). Lexinfo: A declarative model for the lexicon-ontology interface. *Journal of Web Semantics*, 9(1):29–51.

Collins, F. S., Green, E. D., Guttmacher, A. E., and Guyer, M. S. (2003). A vision for the future of genomics research. *Nature*, 422(6934):835–847.

Côté, R. G., Jones, P., Martens, L., Apweiler, R., and Hermjakob, H. (2008). The ontology lookup service: more data and better tools for controlled vocabulary queries. *Nucleic Acids Research*, 36(Web-Server-Issue):372–376.

Coulet, A., Smaïl-Tabbone, M., Benlian, P., Napoli, A., and Devignes, M. (2008). Ontology-guided data preparation for discovering genotype-phenotype relationships. *BMC Bioinformatics*, 9(S-4).

Cruz, I. F., Antonelli Flavio, P., and Stroe, C. (2009). Agreementmaker: efficient matching for large real-world schemas and ontologies. *Proceedings of the Very Large Data Bases Endowment*, 2(2):1586–1589.

d'Aquin, M., Gridinoc, L., Angeletou, S., Sabou, M., and Motta, E. (2007). Watson: A Gateway for Next Generation Semantic Web Applications. In *6th International Semantic Web Conference, ISWC, Poster and Demonstration, Busan, Korea*, pages 11–15.

d'Aquin, M. and Motta, E. (2011). Watson, more than a semantic web search engine. *Semantic Web Journal*, 2(1):55–63.

d'Aquin, M. and Noy, N. F. (2012). Where to publish and find ontologies? A survey of ontology libraries. *Journal of Web Semantics*, 11:96–111.

Darmoni, S. J., Jarrousse, É., Zweigenbaum, P., Beux, P. L., Namer, F., Baud, R. H., Joubert, M., Vallée, H., Côté, R. A., Buemi, A., Bourigault, D., Recourcé, G., Jeanneau, S., and Rodrigues, J. M. (2003). Vumef: Extending the french involvement in the UMLS metathesaurus. In *American Medical Informatics Association Annual Symposium (AMIA), Washington, DC, USA*.

David, J., Euzenat, J., Scharffe, F., and Trojahn dos Santos, C. (2011). The Alignment API 4.0. *Semantic Web Journal*, 2(1):3–10.

Deléger, L., Merkel, M., and Zweigenbaum, P. (2009). Translating medical terminologies through word alignment in parallel text corpora. *Journal of Biomedical Informatics*, 42(4):692–701.

Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R. S., Peng, Y., Reddivari, P., Doshi, V., and Sachs, J. (2004). Swoogle: a search and metadata engine for the semantic web. In *13th ACM International Conference on Information and Knowledge Management, CIKM, Washington, USA*, pages 652–659.

Djeddi, W. E. and Khadir, M. T. (2014). A novel approach using context-based measure for matching large scale ontologies. In *16th International Data Warehousing and Knowledge Discovery Conference, DaWaK, Munich, Germany*, pages 320–331.

Djeddi, W. E., Khadir, M. T., and Ben Yahia, S. (2015). Xmap: results for OAEI 2015. In *10th International Workshop on Ontology Matching, OM, Bethlehem, USA*, pages 216–221.

Do, H.-H., Melnik, S., and Rahm, E. (2002). Comparison of schema matching evaluations. In *Web and Database-Related Workshops, Erfurt, Germany*, pages 221–237.

Donnelly, K. (2006). SNOMED-CT: The advanced terminology and coding system for eHealth. *Studies in health technology and informatics*, 121:279.

dos Santos, C. T., Fu, B., Zamazal, O., and Ritze, D. (2014). State-of-the-art in multilingual and cross-lingual ontology matching. In *Towards the Multilingual Semantic Web, Principles, Methods and Applications*, pages 119–135. Springer.

Dragisic, Z., Ivanova, V., Lambrix, P., Faria, D., Jiménez-Ruiz, E., and Pesquita, C. (2016). User validation in ontology alignment. In *15th International Semantic Web Conference, ISWC, Kobe, Japan*, pages 200–217.

Dragisic, Z., Ivanova, V., Li, H., and Lambrix, P. (2017). Experiences from the anatomy track in the ontology alignment evaluation initiative. *Journal of Biomedical Semantics*, 8(1):56:1–56:28.

Duchateau, F. and Bellahsene, Z. (2016). YAM: A step forward for generating a dedicated schema matcher. *Transactions on Large-Scale Data and Knowledge-Centered Systems XXV*, 25:150–185.

Ehrig, M. and Euzenat, J. (2005). Relaxed precision and recall for ontology matching. In *K-CAP Workshop on Integrating Ontologies, Banff, Canada*, pages 25–32.

Ehrig, M., Sure, Y., and Staab, S. (2005). Supervised learning of an ontology alignment process. In *3rd Biennial Conference on Professional Knowledge Management/Wissensmanagement, WM, Kaiserslautern, Germany*, pages 508–517.

Euzenat, J. (2007). Semantic precision and recall for ontology alignment evaluation, IJCAI, hyderabad, india. In *International Joint Conference on Artificial Intelligence*, pages 348–353.

Euzenat, J. (2008). Algebras of ontology alignment relations. In *7th International Semantic Web Conference, ISWC, Karlsruhe, Germany*, pages 387–402.

Euzenat, J. and Shvaiko, P. (2007). *Ontology Matching*. Springer.

Euzenat, J. and Shvaiko, P. (2013). *Ontology Matching (second edition)*. Springer.

Faria, D., Martins, C., Nanavaty, A., Oliveira, D., Sowkarthiga, B., Taheri, A., Pesquita, C., Couto, F. M., and Cruz, I. F. (2015). AML results for OAEI 2015. In *10th International Workshop on Ontology Matching, OM, Bethlehem, PA, USA*, pages 116–123.

Faria, D., Pesquita, C., Balasubramani, B. S., Martins, C., Cardoso, J., Curado, H., Couto, F. M., and Cruz, I. F. (2016). OAEI 2016 results of AML. In *11th International Workshop on Ontology Matching, OM, Kobe, Japan*, pages 138–145.

Faria, D., Pesquita, C., Santos, E., Cruz, I. F., and Couto, F. M. (2013a). Agreement maker light results for OAEI 2013. In *8th International Workshop on Ontology Matching OM, Sydney, Australia*, pages 101–108.

Faria, D., Pesquita, C., Santos, E., Cruz, I. F., and Couto, F. M. (2014). Automatic background knowledge selection for matching biomedical ontologies. *PloS One*, 9(11):e111226.

Faria, D., Pesquita, C., Santos, E., Palmonari, M., Cruz, I. F., and Couto, F. M. (2013b). The agreementmakerlight ontology matching system. In *On the Move to Meaningful Internet Systems, OTM, Graz, Austria*, pages 527–541.

Farrar, S. and Langendoen, D. T. (2003). A linguistic ontology for the semantic web. *GLOT international*, 7(3):97–100.

Foguem, B. K., Coudert, T., Béler, C., and Geneste, L. (2008). Knowledge formalization in experience feedback processes: An ontology-based approach. *Computers in Industry*, 59(7):694–710.

Fu, B., Brennan, R., and O'Sullivan, D. (2009). Multilingual ontology mapping: Challenges and a proposed framework. In *Workshop on Matching and Meaning-Automated Development, Evolution and Interpretation of Ontologies*, pages 1–31.

Fu, B., Brennan, R., and O'sullivan, D. (2010). Cross-lingual ontology mapping and its use on the multilingual semantic web. In *1st International Workshop on the Multilingual Semantic Web (MSW), Raleigh, North Carolina, USA*, pages 13–20.

Fu, B., Brennan, R., and O'Sullivan, D. (2012). A configurable translation-based cross-lingual ontology mapping system to adjust mapping outcomes. *Journal of Web Semantics*, 15:15–36.

Ghazvinian, A., Noy, N. F., Jonquet, C., Shah, N. H., and Musen, M. A. (2009a). What four million mappings can tell you about two hundred ontologies. In *8th International Semantic Web Conference (ISWC), Chantilly, VA, USA*, pages 229–242.

Ghazvinian, A., Noy, N. F., and Musen, M. A. (2009b). Creating mappings for ontologies in biomedicine: simple methods work. In *American Medical Informatics Association Annual Symposium, AMIA, San Francisco, CA, USA*, pages 198–202.

Giunchiglia, F., Shvaiko, P., and Yatskevich, M. (2004). S-match: an algorithm and an implementation of semantic matching. In *1st European Semantic Web Symposium, ESWS, Heraklion, Crete, Greece*, pages 61–75.

Gracia, J., Montiel-Ponsoda, E., Cimiano, P., Gómez-Pérez, A., Buitelaar, P., and McCrae, J. P. (2012). Challenges for the multilingual web of data. *Journal of Web Semantics*, 11:63–71.

Gracia, J., Montiel-Ponsoda, E., Vila-Suero, D., and Aguado de Cea, G. (2014). Enabling language resources to expose translations as linked data on the web. In *9th International Conference on Language Resources and Evaluation (LREC), Reykjavik, Iceland*, pages 409–413.

Grosjean, J., Merabti, T., Griffon, N., Dahamna, B., and Darmoni, S. (2011). Multi-terminology cross-lingual model to create the european health terminology/ontology portal. In *9th International Conference on Terminology and Artificial Intelligence, Paris, France*, pages 119–122.

Groß, A., Hartung, M., Kirsten, T., and Rahm, E. (2011). Mapping composition for matching large life science ontologies. In *2nd International Conference on Biomedical Ontology, ICBO, Buffalo, NY, USA*, pages 109–116.

Groß, A., Hartung, M., Kirsten, T., and Rahm, E. (2012). Gomma results for OAEI 2012. In *7th International Conference on Ontology Matching, OM, Boston, USA*, pages 133–140.

Gruber, T. R. (1995). Toward principles for the design of ontologies used for knowledge sharing? *International journal of human-computer studies*, 43(5-6):907–928.

Halevy, A. (2005). Why your data won't mix. *Queue*, 3(8):50–58.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18.

Hartung, M., Groß, A., Kirsten, T., and Rahm, E. (2012). Effective mapping composition for biomedical ontologies. In *9th Extended Semantic Web Conference, ESWC, Heraklion, Crete, Greece*, pages 176–190.

Hartung, M., Kirsten, T., and Rahm, E. (2008). Analyzing the evolution of life science ontologies and mappings. In *5th International Workshop on Data Integration in the Life Sciences (DILS), Evry, France*, pages 11–27.

Hecht, T., Buche, P., Dibie, J., Ibanescu, L., and Trojahn dos Santos, C. (2017). *Ontology Alignment Using Web Linked Ontologies as Background Knowledge*, pages 207–227. Springer.

Hoehndorf, R., Schofield, P. N., and Gkoutos, G. V. (2015). The role of ontologies in biological and biomedical research: a functional perspective. *Briefings in Bioinformatics*, 16(6):1069–1080.

Isele, R. and Bizer, C. (2012). Learning expressive linkage rules using genetic programming. *International Journal on Very Large Data Bases*, 5(11):1638–1649.

Ivanova, V. and Lambrix, P. (2013). A unified approach for aligning taxonomies and debugging taxonomies and their alignments. In *10th International Conference the Semantic Web: Semantics and Big Data, ESWC, Montpellier, France*, pages 1–15.

Jain, P., Hitzler, P., Sheth, A. P., Verma, K., and Yeh, P. Z. (2010). Ontology alignment for linked open data. In *9th International Semantic Web Conference, ISWC, Shanghai, China*, pages 402–417.

Jiménez-Ruiz, E., Cuenca Grau, B., Solimando, A., and V. Cross, V. (2015). Logmap family results for OAEI 2015. In *10th International Workshop on Ontology Matching, OM, Bethlehem, PA, USA*, pages 171–175.

Jiménez-Ruiz, E., Grau, B. C., and Cross, V. (2016). LogMap family participation in the OAEI 2016. In *11th International Workshop on Ontology Matching, OM, Kobe, Japan*, pages 185–189.

Jiménez-Ruiz, E., Meilicke, C., Grau, B. C., and Horrocks, I. (2013). Evaluating mapping repair systems with large biomedical ontologies. In *26th International Workshop on Description Logics, Ulm, Germany*, pages 246–257.

Jonquet, C., Annane, A., Bouarech, K., Emonet, V., and Melzi, S. (2016). SIFR bioportal: Un portail ouvert et générique d'ontologies et de terminologies biomédicales françaises au service de l'annotation sémantique. In *16eme Journées Francophones d'Informatique Médicale, JFIM, Geneve, Switzerland.*

Jonquet, C., Emonet, V., and Musen, M. A. (2015). Roadmap for a multilingual bioportal. In *Fourth Workshop on the Multilingual Semantic Web (MSW4) co-located with 12th Extended Semantic Web Conference, ESWC, Portorož, Slovenia.*, pages 15–26.

Jonquet, C. and Musen, M. A. (2014). Gestion du multilinguisme dans un portail d'ontologies: étude de cas pour le ncbo bioportal. In *Terminology and Ontology : Theories and applications Workshop (TOTh), Bruxelles, Belgium*, page 2.

Jonquet, C., Shah, N. H., and Musen, M. A. (2009). The open biomedical annotator. In *American Medical Informatics Association Symposium on Translational BioInformatics (AMIA)*, pages 56–60.

Jonquet, C., Toulet, A., Arnaud, E., Aubin, S., Yeumo, E. D., Emonet, V., Graybeal, J., Laporte, M.-A., Musen, M. A., Pesce, V., and Larmande, P. (2018). Agroportal: A vocabulary and ontology repository for agronomy. *Computers and Electronics in Agriculture*, 144:126 – 143.

Jonquet, C., Toulet, A., and Emonet, V. (2017). Two years later: the landscape of vocabularies and ontologies in the AgroPortal. In *The International Workshop on sources and data integration in agriculture, food and environment using ontologies, IN-OVIVE, Montpellier, France.*

Kensche, D., Quix, C., Li, X., and Li, Y. (2007). Geromesuite: A system for holistic generic model management. In *33rd International Conference on Very Large Data Bases, VLDB, Vienna, Austria*, pages 1322–1325.

Kingkaew, C. (2012). Using Unstructured Documents as Background Knowledge for Ontology Matching. In *International Conference on Machine Learning and Computer Science, IMLCS, Phuket, Thailand*, pages 147–151.

Kirsten, T., Groß, A., Hartung, M., and Rahm, E. (2011). GOMMA: a component-based infrastructure for managing and analyzing life science ontologies and their evolution. *Journal of Biomedical Semantics*, 2:6.

Klein, M. (2001). Combining and relating ontologies: an analysis of problems and solutions. In *Workshop on ontologies and information sharing, Seattle, Washington, USA*, pages 53–62.

Köhler, S. and al. (2014). The human phenotype ontology project: linking molecular biology and disease through phenotype data. *Nucleic Acids Research*, 42(Database-Issue):966–974.

Lambrix, P. and Liu, Q. (2009). Using partial reference alignments to align ontologies. In *6th European Semantic Web Conference, ESWC 2009, Heraklion, Crete, Greece*, pages 188–202.

Lambrix, P. and Tan, H. (2006). SAMBO - A system for aligning and merging biomedical ontologies. *Journal of Web Semantics*, 4(3):196–206.

Lenzerini, M. (2002). Data integration: A theoretical perspective. In *Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 233–246. ACM.

Liang, A. C. and Sini, M. (2006). Mapping AGROVOC and the chinese agricultural thesaurus: Definitions, tools, procedures. *The New Review of Hypermedia and Multimedia*, 12(1):51–62.

Lin, F. and Sandkuhl, K. (2008). A survey of exploiting WordNet in ontology matching. In *2nd International Conference on Artificial Intelligence in Theory and Practice, IFIP, Milan, Italy*, pages 341–350.

Locoro, A., David, J., and Euzenat, J. (2014). Context-based matching: design of a flexible framework and experiment. *Journal on Data Semantics*, 3(1):25–46.

Mary, M., Soualmia, L., and Gansel, X. (2017). Usability and improvement of existing alignments: The LOINC-SNOMED CT case study. In *Knowledge Engineering and Knowledge Management, EKAW, Bologna, Italy*, pages 145–148.

Mascardi, V., Locoro, A., and Rosso, P. (2010). Automatic ontology matching via upper ontologies: A systematic evaluation. *IEEE Transactions on Knowledge and Data Engineering*, 22(5):609–623.

McCrae, J. P., Spohr, D., and Cimiano, P. (2011a). Linking lexical resources and ontologies on the semantic web with lemon. In *8th Extended Semantic Web Conference, ESWC, Heraklion, Crete, Greece*, pages 245–259.

McCrae, J. P., Spohr, D., and Cimiano, P. (2011b). Linking lexical resources and ontologies on the semantic web with lemon. In *8th Extended Semantic Web Conference, ESWC 2011, Heraklion, Crete, Greece*, pages 245–259.

Mehryar, M., Afshin, R., and Ameet, T. (2012). *Foundations of Machine Learning*. Adaptive computation and machine learning. MIT Press.

Meilicke, C., Garcia-Castro, R., Freitas, F., van Hage, W. R., Montiel-Ponsoda, E., de Azevedo, R. R., Stuckenschmidt, H., Sváb-Zamazal, O., Svátek, V., Tamilin, A., dos Santos, C. T., and Wang, S. (2012). Multifarm: A benchmark for multilingual ontology matching. *Journal of Web Semantic.*, 15:62–68.

Miller, G. A. (1995). WordNet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Montiel-Ponsoda, E., Aguado de Cea, G., Gómez-Pérez, A., and Peters, W. (2008). Modelling multilinguality in ontologies. In *22nd International Conference on Computational Linguistics (COLING), Posters Proceedings, Manchester, UK*, pages 67–70.

Mungall, C. J., Torniai, C., Gkoutos, G. V., Lewis, S. E., and Haendel, M. A. (2012). Uberon, an integrative multi-species anatomy ontology. *Genome biology*, 13(1):1–20.

Névéol, A., Grosjean, J., Darmoni, S. J., and Zweigenbaum, P. (2014). Language resources for french in the biomedical domain. In *9th International Conference on Language Resources and Evaluation, LREC, Reykjavik, Iceland*, pages 2146–2151.

Ngo, D. and Bellahsene, Z. (2015). Efficient semantic verification of ontology alignment. In *International Conference on Web Intelligence and Intelligent Agent Technology, WI-IAT, Singapore*, pages 141–148.

Ngo, D. and Bellahsene, Z. (2016). Overview of YAM++:(not) Yet Another Matcher for ontology alignment task. *Journal of Web Semantics*, 41:30 – 49.

Ngo, D., Bellahsene, Z., and Coletta, R. (2011). A flexible system for ontology matching. In *23rd International Conference on Advanced Information Systems Engineering, CAiSE Forum, London, UK*, pages 73–80.

Ngo, D., Bellahsene, Z., and Todorov, K. (2013). Opening the black box of ontology matching. In *10th Extended Semantic Web Conference, ESWC, Montpellier, France*, pages 16–30.

Niu, F., Zhang, C., Ré, C., and Shavlik, J. W. (2012). Deepdive: Web-scale knowledge-base construction using statistical learning and inference. In *Proceedings of the Second International Workshop on Searching and Integrating New Web Data Sources, Istanbul, Turkey, August 31, 2012*, pages 25–28.

Noy, N. F., Griffith, N., and Musen, M. A. (2008). Collecting community-based mappings in an ontology repository. In *7th International Semantic Web Conference (ISWC), Karlsruhe, Germany*, pages 371–386.

Noy, N. F., Shah, N. H., Whetzel, P. L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D. L., Storey, M.-A., and Chute, C. G. (2009). BioPortal: ontologies and integrated data resources at the click of a mouse. *Nucleic Acids Research*, 37:170–173.

O'Leary, D. E. (2005). Review: Ontologies: A silver bullet for knowledge management and electronic commerce. *The computer journal*, 48(4):498.

Otero-Cerdeira, L., Rodríguez-Martínez, F. J., and Gómez-Rodríguez, A. (2015). Ontology matching: A literature review. *Expert Systems with Applications*, 42(2):949–971.

Pedersen, T., Patwardhan, S., and Michelizzi, J. (2004). Wordnet: : Similarity - measuring the relatedness of concepts. In *19th National Conference on Artificial Intelligence, San Jose, California, USA*, pages 1024–1025.

Pesquita, C., Faria, D., Santos, E., and Couto, F. M. (2013). To repair or not to repair: reconciling correctness and coherence in ontology reference alignments. In *8th International Workshop on Ontology Matching, OM, Sydney, Australia*, pages 13–24.

Pratim, T. P. and Koby, C. (2009). New regularized algorithms for transductive learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 442–457.

Quix, C., Gal, A., Sagi, T., and Kensche, D. (2010). An integrated matching system GeRoMeSuite and SMB: results for OAEI 2010. In *5th International Workshop on Ontology Matching, OM, Shanghai, China.*

Quix, C., Roy, P., and Kensche, D. (2011). Automatic selection of background knowledge for ontology matching. In *International Workshop on Semantic Web Information Management, SWIM, Athens, Greece*, pages 5:1–5:7.

Ratner, A., Bach, S. H., Ehrenberg, H. R., Fries, J. A., Wu, S., and Ré, C. (2017). Snorkel: Rapid training data creation with weak supervision. *CoRR*, abs/1711.10160.

Reynaud, C. and Safar, B. (2007). Exploiting WordNet as background knowledge. In *2nd International Workshop on Ontology Matching, OM, Busan, Korea*, pages 291–295.

Romero, M. M., Jonquet, C., O'Connor, M. J., Graybeal, J., Pazos, A., and Musen, M. A. (2017). NCBO ontology recommender 2.0: an enhanced approach for biomedical ontology recommendation. *Journal of Biomedical Semantics*, 8(1):21:1–21:22.

Rong, S., Niu, X., Xiang, E., Wang, H., Yang, Q., and Yu, Y. (2012). A machine learning approach for instance matching based on similarity metrics. In *11th International Semantic Web Conference, ISWC, Boston, USA*, pages 460–475.

Rosse, C. and Mejino, J. L. (2003). A reference ontology for biomedical informatics: the foundational model of anatomy. *Journal of Biomedical Informatics*, 36(6):478 – 500.

Rueda, C., Bermudez, L., and Fredericks, J. (2009). The mmi ontology registry and repository: A portal for marine metadata interoperability. In *OCEANS 2009, MTS/IEEE Biloxi-Marine Technology for Our Future: Global and Local Challenges*, pages 1–6. IEEE.

Sabou, M., d'Aquin, M., and Motta, E. (2006). Using the semantic web as background knowledge for ontology mapping. In *1st International Workshop on Ontology Matching, OM, Athens, Georgia, USA*, pages 1–12.

Sabou, M., d'Aquin, M., and Motta, E. (2008). Exploring the semantic web as background knowledge for ontology matching. *Journal on Data Semantics*, pages 156–190.

Safar, B., Reynaud, C., and Calvier, F. (2007). Techniques d'alignement d'ontologies basées sur la structure d'une ressource complémentaire. In *1ères Journées Francophones sur les Ontologies, JFO, Sousse, Tunisie*, pages 21–35.

Salton, G., Wong, A., and Yang, C.-S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.

Salvadores, M., Alexander, P. R., Musen, M. A., and Noy, N. F. (2013). BioPortal as a dataset of linked biomedical ontologies and terminologies in rdf. *Semantic Web Journal*, 4(3):277–284.

Savenkov, V., Mehmood, Q., Umbrich, J., and Polleres, A. (2017). Counting to k or how SPARQL1.1 property paths can be extended to top-k path queries. In *Proceedings of the 13th International Conference on Semantic Systems, SEMANTICS 2017, Amsterdam, The Netherlands, September 11-14, 2017*, pages 97–103.

Shahri, S. H. and Jamil, H. (2009). An extendable meta-learning algorithm for ontology mapping. In *8th International Conference on Flexible Query Answering Systems, FQAS, Roskilde, Denmark*, pages 418–430. Springer Berlin Heidelberg.

Shamdasani, J., Hauer, T., Bloodsworth, P., Branson, A., Odeh, M., and McClatchey, R. (2009). Semantic matching using the UMLS. In *6th European Semantic Web Conference, ESWC, Heraklion, Crete, Greece*, pages 203–217.

Shvaiko, P. and Euzenat, J. (2008). Ten challenges for ontology matching. In *On the Move to Meaningful Internet Systems, OTM, Monterrey, Mexico*, pages 1164–1182.

Shvaiko, P. and Euzenat, J. (2013). Ontology matching: state of the art and future challenges. *IEEE Transactions on Knowledge and Data Engineering*, 25(1):158–176.

Sioutos, N., de Coronado, S., Haber, M. W., Hartel, F. W., Shaiu, W.-L., and Wright, L. W. (2007). NCI thesaurus: A semantic model integrating cancer-related clinical and molecular information. *Journal of Biomedical Informatics*, 40(1):30 – 43.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., Eilbeck, K., Ireland, A., Mungall, C. J., et al. (2007). The obo foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology*, 25(11):1251–1255.

Song, M., Kim, W. C., Lee, D., Heo, G. E., and Kang, K. Y. (2015). PKDE4J: entity and relation extraction for public knowledge discovery. *Journal of Biomedical Informatics*, 57:320–332.

Spohr, D., Hollink, L., and Cimiano, P. (2011a). A machine learning approach to multilingual and cross-lingual ontology matching. In *10th International Semantic Web Conference, ISWC, Bonn, Germany*, pages 665–680.

Spohr, D., Hollink, L., and Cimiano, P. (2011b). A machine learning approach to multilingual and cross-lingual ontology matching. In *10th International Semantic Web Conference (ISWC), Bonn, Germany*, pages 665–680.

Stoilos, G., Stamou, G., and Kollias, S. (2005). A string metric for ontology alignment. In *4th International Semantic Web Conference, ISWC, Galway, Ireland*, pages 624–637.

Tan, H., Jakonienė, V., Lambrix, P., Aberg, J., and Shahmehri, N. (2006). Alignment of biomedical ontologies using life science literature. In *International Workshop on Knowledge Discovery in Life Science Literature, Singapore*, pages 1–17.

Tchechmedjiev, A., Abdaoui, A., Emonet, V., Melzi, S., Jonnagaddala, J., and Jonquet, C. (2018). Enhanced functionalities for annotating and indexing clinical text with the NCBO annotator+. *Bioinformatics*, 34(11):1962–1965.

Tigrine, A. N., Bellahsene, Z., and Todorov, K. (2015). Light-weight cross-lingual ontology matching with LYAM++. In *On the Move to Meaningful Internet Systems, OTM, Rhodes, Greece*, pages 527–544.

Tigrine, A. N., Bellahsene, Z., and Todorov, K. (2016). Selecting optimal background knowledge sources for the ontology matching task. In *20th International Conference on Knowledge Engineering and Knowledge Management, EKAW, Bologna, Italy*, pages 651–665.

Tordai, A., Ghazvinian, A., Ossenbruggen, J. v., Musen, M. A., and Noy, N. F. (2010). Lost in translation? empirical analysis of mapping compositions for large ontologies. In *5th International Workshop on Ontology Matching, OM, Shanghai, China*, pages 13–24.

Wegner, P. (1996). Interoperability. *ACM Computing Surveys*, 28(1):285–287.

Wijaya, D., Talukdar Partha, P., and Mitchell, T. (2013). Pidgin: ontology alignment using web text as interlingua. In *22nd ACM international Conference on Information & Knowledge Management*, pages 589–598.

Wu, Z. and Palmer, M. (1994). Verbs semantics and lexical selection. In *32nd annual meeting on Association for Computational Linguistics, Las Cruces, New Mexico*, pages 133–138.

Ying, D. and Dieter, F. (2001). Ontology library systems: The key to successful ontology reuse. In *The first Semantic Web Working Symposium, SWWS, Stanford University, California, USA*, pages 93–112.

Zweigenbaum, P., Baud, R., Burgun, A., Namer, F., Jarrousse, É., Grabar, N., Ruch, P., Le Duff, F., Thirion, B., and Darmoni, S. (2003). Towards a unified medical lexicon for french. In *Medical Informatics in Europe (MIE)*.