UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE BIOLOGIA ANIMAL

Compound Matching for Multiple Ontologies

Maria Madalena Ervedosa de Lacerda Pavão

Mestrado em Bioinformática e Biologia Computacional
Especialização em Bioinformática

Dissertação orientada por:
Prof. Doutora Cátia Luísa Santana Calisto Pesquita

2019

Resumo

As áreas científicas multidisciplinares como a Biomédica, usam normalmente redes de ontologias para suportar aplicações como anotação, integração, pesquisa e análise de dados. Estas redes podem ser construídas usando técnicas de correspondência de ontologias, no entanto a maioria das abordagens existentes é limitada a correspondências entre duas ontologias, sendo a grande maioria das equivalências simples. Em cenários de múltiplos domínios, é necessário encontrar correspondências mais complexas, que podem envolver várias ontologias, ou seja, correspondências compostas.

Esta dissertação propõe um novo algoritmo de alinhamentos compostos, capaz de criar correspondências entre uma classe de origem e uma expressão de classe, relacionando múltiplas classes de múltiplas ontologias alvo. Trata das limitações de abordagens anteriores, que apenas consideraram duas classes de duas ontologias alvo. O algoritmo é baseado nas abordagens eficientes de correspondência léxica do AgreementMakerLight.

Uma avaliação automática foi realizada contra alinhamentos de referência parciais usando métricas de avaliação clássicas e também novas, mais adequadas para a avaliação do alinhamento composto. Apesar dos resultados com métricas clássicas serem algo limitados (um facto ao qual não ajuda a incompletude dos alinhamentos de referência), as novas métricas de avaliação, projetadas para medir a utilidade de uma correspondência num cenário de alinhamento interativo, são promissoras, com menor precisão, mas com valores de *recall* entre 80-98%.

Palavras Chave: Alinhamento de Ontologias, Alinhamento Complexo de Ontologias, Alinhamento Composto de Ontologias, Ontologias Biomédicas

Abstract

Multi-domain areas, such as the biomedical field, routinely employ networks of ontologies to support applications such as data annotation, integration, search and analysis. These networks can be built using ontology matching techniques, however most existing approaches are limited to matches between two ontologies, the large majority being simple equivalences. In multi-domain scenarios, there is a need to discover more complex mappings, that may involve multiple ontologies, i.e. compound mappings.

This thesis proposes a novel compound matching algorithm, able to compose mappings between a source class and a class expression relating multiple classes from multiple target ontologies. It addresses the limitations of previous approaches that only considered two target classes from two target ontologies. The algorithm is based on the efficient lexical matching approaches in AgreementMakerLight.

An automatic evaluation was carried against partial reference alignments using both classical and novel evaluation metrics more suited to compound alignment evaluation. Despite results with classical metrics being rather poor (a fact not helped by the incompleteness of the reference alignments), the novel evaluation metrics, designed to measure the usefulness of a mapping in an interactive alignment scenario are promising, with lower precision, but recall values in the 80-98% range.

Keywords: Ontology Matching, Complex Ontology Matching, Compound Ontology Matching, Biomedical Ontologies, Semantic Data Integration

Resumo Alargado

As ontologias são teorias sobre os objetos e as relações entre eles, descrevendo o conhecimento acerca de um domínio. As ontologias diferem entre si, mas têm em comum a premissa de que existem objetos e que estes têm propriedades ou atributos a que podem ser imputados valores. Os objetos, por seu lado, podem dividir-se em partes e podem também combinar-se entre si através de várias relações. Mais formalmente, as ontologias podem ser descritas como vocabulários específicos de um domínio com um conjunto de entidades, os seus atributos e as relações entre elas, para além de restrições, regras e axiomas. A necessidade de uniformizar o vocabulário específico de um domínio não é recente, e foi nos anos 70 que foi reconhecida a importância de padronizar o conhecimento em Ciências da Computação, de modo a que possa ser interpretado por computadores. Isto levou a que as ontologias se tornassem uma ferramenta que permite que programas interajam diretamente entre si, tornando possível a partilha e o raciocínio computacional. Assim, as ontologias possibilitam a partilha de um entendimento comum em relação a um domínio e a partilha de conhecimento, para além de permitirem criar novo conhecimento sobre o domínio.

O volume e a complexidade dos dados gerados em Ciências da Vida têm aumentado massivamente nos últimos anos, levando a uma necessidade de gerir, integrar e analisar os dados disponíveis. As ontologias tornaram-se cada vez mais comuns e bem-sucedidas na área Biomédica precisamente por esta ser uma área tão vasta e heterogénea em dados produzidos. Ligar dados biomédicos a ontologias é uma solução promissora para os desafios de integrar, procurar, obter e resolver ambiguidade dos dados, revolucionando a tradicional investigação biomédica ao permitir a partilha e o reconhecimento comum de dados entre a comunidade científica. O alinhamento de ontologias é um processo complexo que resulta numa série de correspondências entre classes de duas ontologias, que vai permitir lidar com a heterogeneidade

semântica, especialmente na área Biomédica, onde muitas vezes as ontologias têm sobreposições entre si. A maioria das abordagens de alinhamento de ontologias focam-se em correspondências binárias, de um para um, mas abordagens mais recentes têm-se focado em alinhamentos compostos, entre mais do que duas ontologias.

O trabalho desenvolvido no âmbito desta dissertação é baseado numa abordagem de *longest word sequence*, e tem dois pontos principais: a construção de um léxico de conjuntos de palavras formados a partir dos conceitos da ontologia e um algoritmo de filtragem do alinhamento produzido por *matchers* léxicos do *AgreementMakerLight*. O léxico de conjuntos de palavras desenvolvido armazena todas as combinações de palavras sequenciais de uma *label* de uma classe de uma ontologia. Para além de guardar as combinações de palavras, guarda também, para cada uma, um valor correspondente à cobertura destas, ou seja, a razão entre o número de palavras no conjunto e o número de palavras na *label* original. O algoritmo criado, que encontra as correspondências, é baseado numa procura linear das entradas no léxico de conjuntos de palavras da ontologia de fonte sobre o léxico normal do *AgreementMakerLight* contruído para as ontologias alvo. Todas as correspondências são guardadas numa lista de correspondências parciais, às quais é atribuída uma medida de semelhança. Para cada classe de fonte há uma lista de correspondências parciais, cada uma equivalendo a diferentes combinações de palavras. A correspondência é construída ao selecionar uma combinação apropriada de correspondências parciais. Posteriormente, um algoritmo de seleção começa por ordenar as correspondências parciais em ordem decrescente de semelhanças. Para cada classe de fonte, o algoritmo itera sobre as correspondências parciais e adiciona-as a uma correspondência intermédia se se verificar que o conjunto de palavras ao qual foram mapeadas estiver contido na *label* corrente e se o conjunto de palavras ao qual foram mapeadas já não tiver uma correspondência de maior semelhança. Finalmente, uma semelhança final é calculada para esta correspondência como sendo a média de todas as semelhanças de cada correspondência parcial.

A avaliação da abordagem foi efetuada utilizando oito conjuntos de teste.

As ontologias de fonte utilizadas foram a *Human Phenotype Ontology* e a *Mammalian Phenotype Ontology*. Para cada uma destas, os alvos eram os conjuntos das ontologias *Uber Anatomy Ontology* (UBERON) e *Phenotypic Quality Ontology* (PATO) (dois conjuntos HP-UB-PT e dois MP-UB-PT) ou das ontologias UBERON, PATO e *Gene Ontology* (GO) (dois conjuntos HP-UB-PT-GO e dois MP-UB-PT-GO). A um conjunto de cada foi aplicado um algoritmo de *stemming*, cujo objetivo é reduzir cada uma das palavras de cada termo das ontologias ao seu radical, formando assim os oito conjuntos. Para a automatização da avaliação, foram gerados quatro ficheiros de referência, recorrendo ao axioma de *Equivalent Classes* das ontologias HP e MP em formato OWL. Os quatro ficheiros de referência correspondiam aos quatro conjuntos distintos descritos anteriormente.

A primeira abordagem para avaliar os resultados consistiu numa classificação das correspondências em comparação com os ficheiros de referência produzidos. Verificou-se que um máximo de 12.9% de correspondências eram exatamente iguais aos ficheiros de referência.

Considerando que se verifica que uma correspondência parcial também pode ser útil, fez-se uma avaliação permissiva dos resultados em que se consideraram como corretas, não só as correspondências iguais às referências, mas também as correspondências que estavam contidas nas referências, as que continham as referências e também as que fossem diferentes mas que tivessem termos sobrepostos, ou seja, que tivessem pelo menos uma classe correspondida na referência. Esta forma de cálculo dos verdadeiros positivos melhorou as métricas. A precisão variou entre 3.4% e entre 35.4% (e entre 3.5% e 38.2% para os resultados *stemmed*), enquanto que o *recall* variou entre 72% e 96.5% (e entre 79.7% e 98% para os resultados *stemmed*), fazendo com que a *f-measure* variasse entre 6.6% e 47.5% (e entre 6.6% e 52.3% para os resultados *stemmed*).

A uma dada altura do processamento do algoritmo são encontradas várias combinações de correspondências, ordenadas de forma decrescente de semelhança. Estas combinações foram consideradas para verificar se a correspondência correta (igual à referência) se encontrava presente nas três ou

cinco primeiras correspondências, em vez de na primeira. De facto, uma quantidade de correspondências corretas não chegava ao alinhamento final. Contando com estas correspondências, verificou-se um aumento das métricas de precisão, *recall* e *f-measure*, de 2% em média, quando comparadas com as métricas normais (não permissivas). Quando se lida com grandes ontologias e com algoritmos que realizam buscas lineares, é esperado que o tempo de corrida do algoritmo seja elevado. Em comparação com outras abordagens, no algoritmo desenvolvido o tempo de corrida é bastante reduzido, mesmo quando a Gene Ontology está em jogo e o tempo de corrida aumenta.

Criar alinhamentos de correspondências compostas é uma tarefa complexa, dado que é necessário encontrar a melhor correspondência entre imensas classes de várias ontologias diferentes. As métricas clássicas de precisão, *recall* e *f-measure* verifica-se não serem as mais adequadas para avaliar este tipo de alinhamentos, uma vez que se baseiam na contagem de verdadeiros positivos e foi considerado que, mesmo que uma correspondência não esteja completamente correta, pode ainda ser útil.

Em comparação com outras, nesta abordagem não é necessário ter em conta a ordem pela qual as ontologias alvo são introduzidas, mesmo quando são utilizadas mais do que duas, pois o algoritmo cria várias correspondências possíveis e escolhe a melhor, independentemente da ordem das ontologias alvo. Permite alinhamentos mais completos ao não restringir o número de termos correspondidos por ontologia alvo, sem comprometer a relevância dos resultados obtidos, tendo sido obtidos valores de recall entre os 80 e os 98%. Apesar desta busca alargada, o tempo de corrida é reduzido, tendo demorado para a tarefa mais morosa menos de cinco minutos.

Acknowledgements

I would like to start by expressing my gratitude to my supervisor, Prof. Cátia Pesquita, for her guidance and knowledge sharing throughout this entire process and for all the patience and encouragement given to me. I appreciate the opportunity and the trust placed in me from the beginning.

I would also like to thank to Fundação da Ciência e Tecnologia for the nine month research grant through funding of LaSIGE Research Unit, UID/CEC/00408/2013 and by the project SMiLaX (PTDC/EEI-ESS/4633/2014).

To my friends from the Master's, thank you for all the lunches and spirit-lifting coffee breaks (and for all the shared Kinder Buenos). To Sofia and Tiago, with whom I shared every day of my time in LaSIGE, making it a great time. To Ana Sofia, who was always by my side during this phase and who always helped me move forward with her example of persistence and strength.

To Joana Teixeira and Catarina Cardoso, for their advice and examples, which helped me keeping focused. To my old friends and my new friends.

Finally, I would like to thank to my family. They have supported me throughout my life, giving me the best tools and advice to make my own decisions and get where I am today. To my grandmother and my uncle, thank you for always having kind and friendly words to say to me. To my little brother I thank for his companionship, for being an example of a person and for making my life more complete. I hope this work can be somehow a guidance for him. At last, to my mother and my father for their unconditional love, support and help and from whom I have learned so much. Their example is what makes me want to improve myself every day.

"... when you connect data together, you get power."

Sir Tim Berners-Lee

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

Biomedical scientists are producing and recording enormous quantities of data everyday, due to high-throughput molecular biology studies and also to the increasingly widespread use of health informatics and electronic health records. This information is stored in databases, being it a structured or unstructured one, and most of the knowledge acquired through data analysis is documented in scientific papers or other forms of natural language, making the use of that knowledge to both humans and machines a challenge, as well as making interoperability between biomedical databases defying (Smith *et al.*, 2007).

Linking biomedical data to ontologies is a promising solution for these challenges of integrating, searching, retrieving and resolving ambiguity of data, revolutionizing the traditional biomedical research by empowering the sharing of data amongst the scientific community. Ontologies bring a common vocabulary, as they describe the semantics of the terms used in the domain. This way, ontologies allow researchers to better capture hidden knowledge from large amounts of original data.

However, linking data to a single ontology is not sufficient in most cases, given that biomedical research commonly spans multiple domains and topics. For instance, describing a patient's record may include using SNOMED-CT for clinical methods employed, LOINC for laboratory analyses and results, ICD-10 for diagnoses and ATC for coding any precribed antibiotics. If more than one ontology is necessary to accurately describe and link the data, to allow true interoperability there is the need to establish links between the multiple ontologies. However, current ontology matching techniques are mostly devoted to finding links between two equivalent entities from two distinct

ontologies. When dealing with more complex domains, it may be necessary to establish more complex mappings or even link more than two ontologies. Complex matching, i.e., finding correspondences that go beyond equivalence between two ontology entities and are able to capture more complex relationships between entities or sets of entities, is a recognized challenge. An example of a complex mapping could be the mapping between the concept "AcceptedPaper" in one ontology, to the entity "Paper" in a second ontology, which has the associated property "Accepted" (Ritze *et al.*, 2009). However, in multi-domain areas, such as epidemiology, healthcare or translational biomedicine, there is a need to link multiple ontologies to address different perspectives on the underlying data(Ferreira *et al.*, 2012), while maintaining the the inherently distributed paradigm championed by the Semantic Web (Berners-Lee *et al.*, 2001). This need motivates another type of complex mappings - 'compound mappings', i.e. matches between class or property expressions involving more than two ontologies. A specific case is the ternary compound matching, (Oliveira & Pesquita, 2018) whereby two classes are related to form a class expression that is then mapped to a third class. For instance, the Human Phenotype Ontology (HP) class HP:0000337 labelled "broad forehead" is equivalent to an axiom obtained by relating the classes PATO:0000600 ("increased width") and FMA:63864 ("forehead"), from the Phenotypic Quality Ontology (PATO) and the Foundation Model of Anatomy (FMA) ontologies respectively, via an intersection. Such mappings allow a fuller semantic integration of multidimensional semantic spaces, supporting more complex data analysis and knowledge discovery tasks.

Compound matching need not be limited to ternary mappings, and may in fact involve multiple concepts from multiple ontologies. This poses additional challenges, related both to the inherently more difficult task of composing a mapping using an arbitrary number of concepts coming from multiple ontologies, but also to the computational complexity behind the task given the large size of biomedical ontologies and their complex and rich vocabularies.

1.1 Objectives

The main objective of this work was to develop a novel approach for compound ontology matching that is able to establish mappings between a class in a source ontology and

any number of classes from a selected set of target ontologies. This approach needs to address the challenges of semantic complexity, lexical variability and ontology size.

These issues were addressed by exploring the computational efficient approaches for purely lexical matches developed by AgreementMakerLight (Faria *et al.*, 2013), and a lexical-based approach to select the best target classes combination.

Additionally, this work also addresses the challenge of evaluating compound alignments, by proposing alternative definitions for well-known performance metrics that are able to produce more useful outputs based on lexical evaluations.

1.2 Contributions

This work has produced several contributions:

- a novel compound matching approach for aligning a source ontology to multiple target ontologies
- the implementation of these algorithms in the open-source ontology matching system AML
- a novel approach for evaluating compound alignments based on a lexical evaluation of mappings.
- a poster in The Thirteenth International Workshop on Ontology Matching[1] titled "Complex matching for multiple ontologies: an exploratory study".

1.3 Overview

This dissertation is divided into six different chapters.

The present Chapter is a contextualization of this dissertation and presents the motivations and objectives, as well as the contributions of the developed work.

Chapter 2, *Concepts and Related Work*, presents some notions relevant to this dissertation about ontologies, more specifically of the biomedical domain, as well as some related work in the areas of complex and compound matching.

Chapter 3, *Methods*, presents the complex matching approach, detailing the algorithms of the new lexicon and of the filter.

[1] http://om2018.ontologymatching.org/

Chapter 4, *Evaluation and Results* details the evaluation of the approach, namely, the construction of reference alignments, the evaluation metrics employed and the performance obtained in different alignment tasks.

After presentation of the results in the previous chapter, Chapter 5, *Discussion*, discusses the obtained results from the automatic evaluation performed against the reference alignments, referring to each aspect of the obtained results.

Chapter 6, *Conclusions and Future Work* concludes the work of this dissertation with some remarks on this exploratory study and avenues for future research.

Chapter 2

Concepts and Related Work

This chapter describes a few concepts necessary to contextualize this work, namely biomedical ontologies and ontology matching, as well as some related work in complex matching.

2.1 Biomedical Ontologies

Ontology is, in philosophy, the study of things that exist. In computer science, the most popular definition of ontology was proposed by Gruber in the early 1990's: "an explicit specification of a conceptualization" (Gruber, 1993). In both cases, ontologies are theories about objects and the relations among them in a given domain, describing the knowledge about the domain. Ontologies differ among them but there is a general agreement on several points, such as *that there are objects in the world*, *objects have properties or attributes that can take values*, *objects can exist in various relations with each other* and *objects can have parts* (Chandrasekaran *et al.*, 1999).

The need for standardizing domain vocabulary is not recent and it was around the 70's when the need to standardize knowledge in computer science in a manner that could be read by computers was recognized. This lead to the development of ontologies as a tool that enables programs to interact directly with information produced by other programs, allowing information sharing and computer reasoning. Tim Berners-Lee took these ideas into World Wide Web, turning it into the Semantic Web, where Internet servers are able to interoperate with each other and build upon each other's data

(Robinson & Bauer, 2011). This made ontologies become part of the W3C standards[1] for the Semantic Web, as one of the constituents of Semantic Technologies, considering that ontologies bring a way to link pieces of information together on the Web of Linked Data (Ontotext, 2019).

More formally, ontologies can be described as domain-specific vocabularies with a set of entities, their attributes and relations, which describe interactions between the entities, as well as restrictions, rules and axioms (Ontotext, 2019). Thus, ontologies allow the sharing of common understanding of the knowledge regarding a specific domain and also the reuse of knowledge. They make domain assumptions explicit, avoiding ambiguity even between generic and shared concepts (Euzenat *et al.*, 2007).

Structure-wise, ontologies are usually organized as Directed Acyclic Graphs (DAG), where the graph nodes are entities and the edges are the relationships (links) between them. In figure 2.1 can be seen an excerpt of the Gene Ontology (GO), showing the ancestry for the term GO:0005739 - mitochondrion.

Subsequently, besides introducing shareable and reusable knowledge, ontologies also allow to add new knowledge about the domain they represent by expressing relationships and enabling the linking of concepts in a variety of ways, unlike other methods with formal specifications of knowledge, such as vocabularies, taxonomies, thesauri and logical models (Ontotext, 2019).

The volume and complexity of data generated in the past years in the field of life sciences has massively increased, leading to the need for managing, integrating and analysing the available data. In the "post-genomic era", the focus of the biomedical community has shifted from just making new discoveries to deal with the information resultant of the genomic research (Bodenreider *et al.*, 2005). The start of the use of ontologies in this field dates to 1998 with the development of one of the most known biological ontologies, the Gene Ontology (GO) (Ashburner *et al.*, 2000).

From there, ontologies became more common and successful in the biomedical field due to its characteristics, namely the large quantity and heterogeneity of produced data. This is exactly what ontologies aim to solve, by providing standard identifiers for classes and relations, representing phenomena within a domain; providing a vocabulary;

[1] https://www.w3.org/standards/semanticweb/

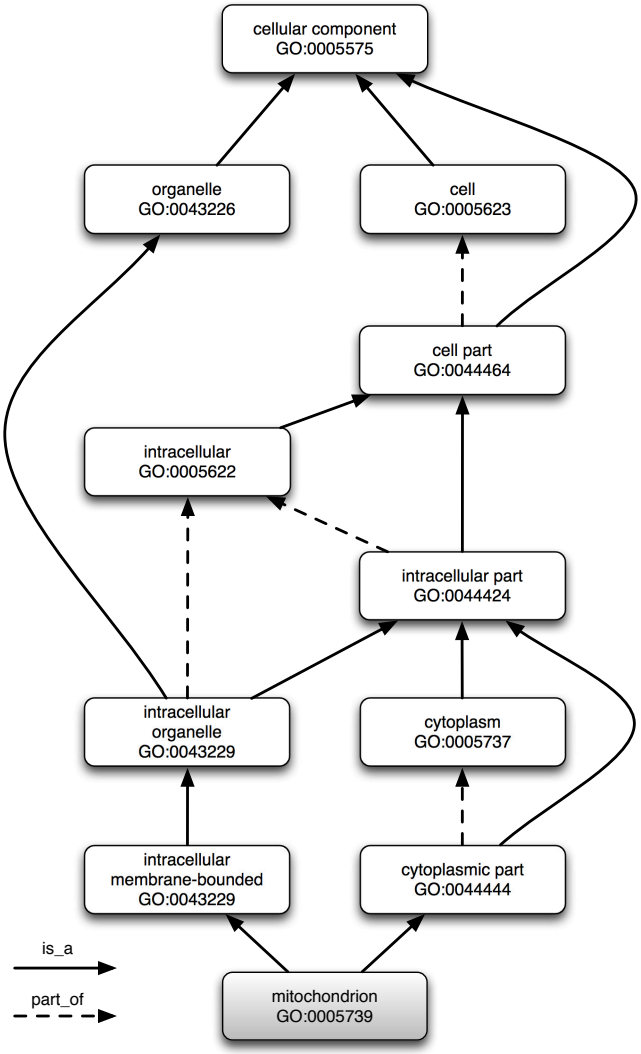

Figure 2.1: Graph representation of Gene Ontology adapted from QuickGO.[1]

providing metadata; providing machine-readable axioms and definitions, allowing computational access to some aspects of the meaning of classes and relations (Hoehndorf *et al.*, 2015). Hence, the main characteristics of biomedical ontologies are:

- **Large size**: biomedical ontologies usually have thousands of classes, or more, which can be computationally challenging.
- **Complex vocabulary**: biomedical ontologies encode several names for the same

[1] `https://www.ebi.ac.uk/QuickGO/GTerm?id=GO:0005739`

class, including a main label and several synonyms.

- **Rich axioms**: biomedical ontologies establish different kinds of relations between classes, leading to a greater semantic richness.

Biomedical ontologies can be tools for annotation and data integration, facilitating the communication of results among communities of scientists. In a way, they can standardize the research findings. On another hand, they can be used in the development of bioinformatics tools with several endings, such as analysis of microarray data and network modelling. More specifically in health care, ontologies can be used in knowledge-based systems like decision support, highly dependent on large amounts of domain knowledge (Musen *et al.*, 2014).

Bioportal (Whetzel *et al.*, 2011), developed by the National Center for Biomedical Ontology (NCBO), is nowadays the largest repository of biomedical ontologies, comprising over 768 ontologies at this date. In this repository there are ontologies in several formats, being the two main OWL and OBO.

The Web Ontology Language (OWL) is a Semantic Web language designed by the W3C as a computational logic-based language, allowing computer programs to exploit knowledge. OWL documents, or ontologies, are Resource Description Framework (RDF) graphs, i.e., a set of RDF triples. (McGuinness *et al.*, 2004). An OWL ontology is composed of Individuals, the objects of the domain, Properties, relations among the individuals, and Classes, groups of individuals that share something in common. (Horridge *et al.*, 2009). On the other hand, the Open Biomedical Ontologies (OBO) language has a simpler format and is more human readable than OWL. An OBO ontology is composed of a set of *stanzas*, each of them describing one element in the ontology, which have an unique identifier and a human readable description. The rest of the stanza consists on a series of tags representing other properties of the element (Golbreich *et al.*, 2007). Although these are different representations of the same domain, the semantic meaning is the same among formats.

Essentially, the existence of distinct modelings of the same domain obeys to the natural human instinct to have different perspectives and hence to model problems differently. When these domains are represented using ontologies, the solution typically involves the use of ontology matching techniques to solve the problem of semantic heterogeneity. Ontologies and ontology matching techniques are an increasing trend

as ontologies provide probably the most interesting opportunity to encode meaning of information. The last decades have born witness to a period of extensive research in this field (Otero-cerdeira *et al.*, 2015).

2.2 Ontology Matching

Ontology matching is a complex process that results in the generation of an alignment, i.e., a series of correspondences between ontology classes (Euzenat *et al.*, 2007; Thiéblin *et al.*, 2018a). This alignment is the culmination of the main goal of ontology matching, which is to deal with the semantic heterogeneity of ontologies when used by computer systems (Euzenat *et al.*, 2007). Especially in the biomedical field, ontologies often overlap and the process of ontology matching can help reducing the semantic gap between ontologies of the same domain (Otero-cerdeira *et al.*, 2015).

More formally, the matching process can be defined as a function f that returns an alignment *A'* from a pair of ontologies *o* and *o'* (Euzenat *et al.*, 2007), process that can be extended with an input alignment A and other parameters p, such as weights and thresholds, and external knowledge r:

$$A' = f(o, o', A, p, r)$$

The process of ontology matching is performed by algorithms called matchers. Matchers use different strategies to compute similarity between two different ontology classes, and when the similarity is associated to the classes, a mapping, i.e., a correspondence, is created. Matchers usually try to find equivalence mappings, in which two entities from different ontologies that represent the same concept, but there are other kinds of mappings, such as mappings of consequence, subsumption and disjointness.

Most of the approaches focus on generating binary correspondences, i.e., one entity of one ontology linked to one entity of another ontology. However, these simple matches are not sufficiently meaningful to entirely overcome ontology heterogeneity, requiring the relationships between the entities to be more expressive. For this, complex, or compound, ontology matching approaches generate mappings between entities of more than two ontologies, better expressing relationships between them (Thiéblin *et al.*, 2018a).

Pesquita *et al.* (2014) define a ternary compound mapping as a tuple $<X, Y, Z, R, M>$, where X, Y and Z represent classes from three different ontologies. The relation R be-

tween Y and Z generates a class expression, which is mapped to X through a mapping relation M. Here X is considered as a source ontology and Y and Z as target ontologies. It is possible to broaden this vision, saying that a compound mapping is a tuple $<C_s,[C_{t0},...,C_{tn}],[P_{t0},...,P_{tn}],M>$, where C_s is a class from a source ontology, $[C_{t0},...,C_{tn}]$ and $[P_{t0},...,P_{tn}]$ are a set of target classes extracted from multiple target ontologies and a set of properties that related each target class to the others, while M is a mapping relation established between the source class and the set of target classes. An example of a compound mapping can be seen in 2.2.

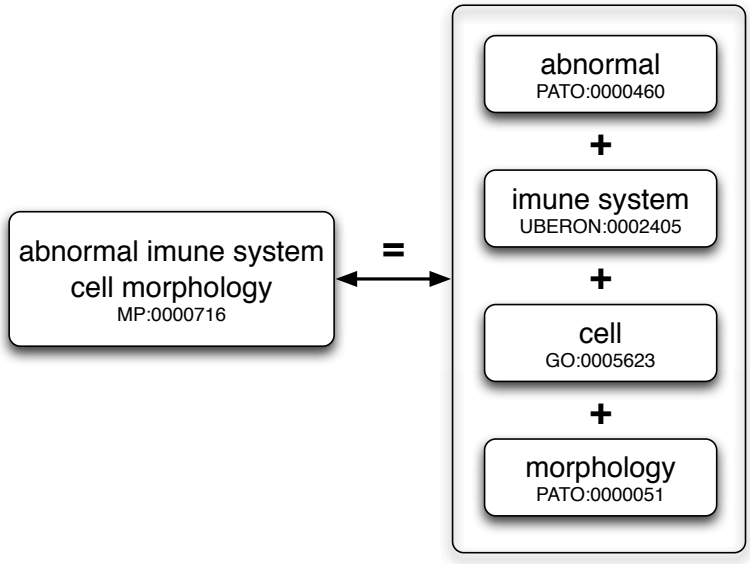

Figure 2.2: Example of a possible complex mapping between the Mammalian Phenotype, Uberanatomy, Phenotypic Quality and Gene Ontology ontologies.

2.3 Ontology Matching Tools

There are over 60 different ontology matching systems (Otero-cerdeira *et al.*, 2015), of which several still receive updates and enhancements frequently. In this section, five popular and high-performance systems, following recent results of Ontology Alignment Evaluation Initiative, will be described.

2.3.1 AgreementMaker

Developed by Cruz *et al.* (2009), AgreementMaker is a schema and ontology matching tool that comprises several kinds of matchers. This tool allows for different inputs, which can be not only ontologies but also the output of the application of one or more matchers. It is also possible to combine several matchers using the same input, combining the results after. What differentiated this tool from others at the time of its development was the fact that it integrates the evaluation of the quality of the mappings with a sophisticated graphical user interface. Therefore this is a very versatile tool with a flexible and extensible framework.

2.3.2 AgreementMakerLight

Derived from AgreementMaker, AgreementMakerLight (AML) was developed by Faria *et al.* (2013) as a novel ontology matching tool, able to handle very large ontologies (with more than thousands of concepts). Similarly to AgreementMaker, AML is also a flexible and an easy to extend tool and has been in continuous development since it was created. One of these extensions was developed by Oliveira & Pesquita (2015) to allow for compound matching. AML is currently the best performing system for Biomedical ontology matching.

2.3.3 LogMap

LogMap is a tool developed by Jiménez-Ruiz & Cuenca Grau (2011), based on an initial set of anchor mappings (i.e., 'almost exact' lexical correspondences), produced from structures that keep lexical and structural information. Starting from the initial anchors and using the ontologies' extended class hierarchy, the algorithm alternates between mapping repair and mapping discovery steps. LogMap has also an ontology reasoner and a greedy diagnosis algorithm.

Since its creation in 2011, some variations were introduced: LogMapLt, the lightweight variant, which applies only string matching techniques, and LogMapBio which includes an extension to use BioPortal as a dynamic provider of mediating ontologies instead of relying on a few preselected ontologies (Jiménez-Ruiz *et al.*, 2016).

2.3.4 XMap

The algorithm proposed by Djeddi & Khadir (2010) takes advantage of features of the OWL language to deduct the similarity between ontology entities. It exploits the common linguistic and structural elements between the entities to measure the similarity between two OWL classes.

2.3.5 YAM++

The YAM++ tool proposed by Ngo & Bellahsene (2012) as an extension of the originally developed tool, YAM (Yet Another Matcher) (Duchateau *et al.*, 2009), uses machine learning techniques to match ontologies when learning data is available. If this is not the case, YAM++ uses textual features of ontologies to provide similarity metrics. It uses element and structural level matchers to discover new mappings, which are then revised by a semantical matcher in order to remove inconsistencies. YAM++ is also able to deal with multilingual ontology matching by first discovering the language in which the ontology is and then translating all labels to English with Microsoft Bing Translator tool.

2.4 Related Work

Compound matching is closely related to complex matching. There have been several works in the area of complex ontology matching, which is commonly described as a correspondence between two classes from two different ontologies, where one of them is a complex concept or property description. The alignment involves only two ontologies, but each mapping contains more than two entities in those ontologies. An example of a complex mapping could be the alignment of the concept "AcceptedPaper" in one ontology, to the entity "Paper" in a second ontology, which has the associated property "Accepted" (Ritze *et al.*, 2009).

One of the first works mentioning the need for complex matching was the one of Maedche *et al.* (2002), where the authors proposed the tool MAFRA for complex ontology matching.

Thiéblin *et al.* (2018b) divided complex matching approaches into four categories: (1) pattern-based with no instance data: Ritze *et al.* (2009) and Ritze *et al.* (2010); (2) pattern-based with instance data: Bayes-ReCCE (Walshe *et al.*, 2016), Parundekar

et al. (2010) and Parundekar *et al.* (2012); (3) non-pattern based with instance data: Nunes *et al.* (2013) and Qin *et al.* (2007); (4) non-pattern based with no instance data: KAOM (Jiang *et al.*, 2016).

In a survey performed by the same authors (Thiéblin *et al.*, 2018a) is proposed another classification of the complex matching approaches. It is based on the specificities of the approaches, namely the type of correspondences (the output), and the structure used to guide the correspondence detection. In terms of type of correspondence, it can be of: logical relations (Bayes-ReCCE (Walshe *et al.*, 2016), CGLUE (Doan *et al.*, 2003)), transformation functions (COMA++ (Arnold, 2013), Nunes *et al.* (2013)) and blocks ((Hu & Qu, 2006)). Regarding the guiding structures, the approaches can be divided into atomic patterns ((Scharffe, 2009), (Ritze *et al.*, 2009), (Ritze *et al.*, 2010)), composite patterns (CGLUE (Doan *et al.*, 2003), COMA++ (Arnold, 2013)), path ((Hu *et al.*, 2011a), (Dou *et al.*, 2010)), tree (MapOnto (An *et al.*, 2005b), (An *et al.*, 2005a)) and finally, no structure ((Hu *et al.*, 2011b), (Hu & Qu, 2006)).

There are fewer works in the area of compound matching. Compound ontology matching is a relatively recent concept, first introduced by Pesquita *et al.* (2014). They proposed a way to create benchmarks to test the performance of matching systems, for which they used OBO cross products to create these reference compound alignments. Besides that, they also developed a strategy for compound matching, integrated in AML. First it matches the source ontology to each of the target ontologies individually, with a using an 'anchor'-based word-matching algorithm, and then matches only all pairs of target classes that map individually to the same source class. The way they measured similarity between source and target was by employing a modified Jaccard index. This strategy lowered the search spaced but it was still not enough to be successful when employed to larger sets of ontologies.

Oliveira & Pesquita (2018) developed ternary compound matching algorithms able to find mappings between a class and a class expression built by the intersection of two classes from two different ontologies. The algorithm was also developed within AML and starts by performing a pairwise mapping of the labels of the source ontology with the labels of the target ontology to match first (target 1) and a similarity is calculated for each mapping. After, a filter is applied to remove all mappings with similarity below a threshold and removes all the source classes which were not mapped to any target 1 classes. It also reduces the number of words of the source labels by

removing from the mapped classes all the words that had a match with a word from a target 1 class. For each of the remaining mappings, the algorithm performs a pairwise mapping of the reduced source labels against the labels of the last target (target 2) and a final similarity is computed for each mapping. Some of the limitations of this work, namely the constraint imposed by using just two classes from two ontologies to build the compound equivalent class, and the necessity of identifying one ontology as the main target, inspired the techniques proposed in this paper.

Chapter 3

Methods

A compound alignment task receives as input a source ontology and a set of target ontologies and produces mappings between classes in the source ontology and class expressions obtained by combining classes from the target ontologies. For the purpose of this dissertation, the approach is restricted to finding mappings where the relation between classes is one of equivalence, and simplified by just finding the set of target classes to map to the source without discovering the accompanying properties.

3.1 Compound Matching Approach

This compound matching approach is based on two main steps: (1) building the internal structures to support the matching algorithm, i.e. building the word sequence lexicon and the lexicon; (2) the matching algorithm itself.

The compound matching algorithm is based on a longest word sequence mapping approach. It is an entirely lexical approach that takes the labels (main and synonyms) of each source class and finds partial lexical matches between word sequences in the source class labels and full labels of target classes. A greedy approach is then applied to select the longest word sequences that provide the highest coverage of a given source label.

The approach for compound matching was developed within the AgreementMakerLight (AML) system (Faria *et al.*, 2013). AML is an easy to extend ontology matching system, that includes several capabilities such as using external knowledge and performing alignment logical repair.

3.1.1 Lexicon

AML uses hashmap based structures to store the lexical information (i.e., the labels and synonyms of each entity) of the ontologies, which are called Lexicons. There are two lexicons involved in this compound alignment strategy: a word sequence lexicon that is created for the source ontology, and a lexicon that is created for all target ontologies.

The Word Sequence Lexicon (WordSeqLexicon), developed in the scope of this dissertation, is a structure that stores all word sequence combinations for each ontology label, and is an extension of AML's standard Lexicon. As an illustration, let's take ontology term "Duct Salivary Gland System". It will correspond to one entry in the Lexicon, "Duct Salivary Gland System", and to 10 entries in the WordSeqLexicon: "Duct", "Duct Salivary", "Duct Salivary Gland", "Duct Salivary Gland System", "Salivary", "Salivary Gland", "Salivary Gland System", "Gland", "Gland System", and "System". In the WordSeqLexicon, for each word sequence, it stores the corresponding *coverage* as well, i.e., the ratio between the number of words in the word sequence and the number of words in the original label (1/4 for "Duct", 2/4 for "Duct Salivary", and so on), as per Figure 3.1.

$$coverage = \frac{\text{words in word sequence}}{\text{words in term}}$$

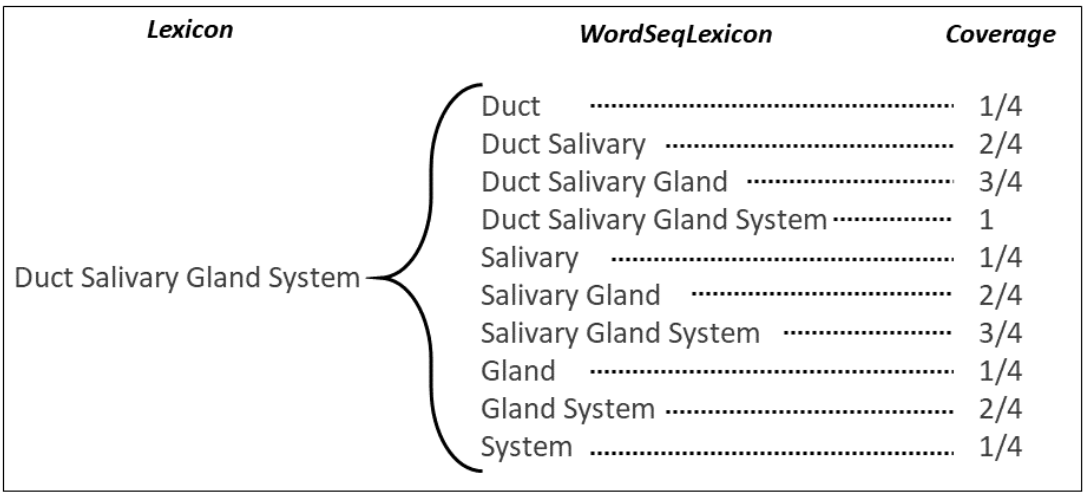

Figure 3.1: Illustration of the Word Sequence Lexicon algorithm.

The algorithm used to create the WordSeqLexicon is detailed in Algorithm 1.

Algorithm 1 WordseqLexicon algorithm

```
1: procedure BUILDWORDSEQLEXICON
2:     entities ← list of classes in source ontology
3:     for entity in entities do
4:         names ← list of labels for entity
5:         for name in names do
6:             wordseqs ← STRINGCOMBINATIONS(name)
7:             for wordseq in wordseqs do
8:                 if wordseq equals name then
9:                     coverage ← 1
10:                else
11:                    coverage ← LENGTH(wordseq)/LENGTH(name)
12:                end if
13:                WordSeqLexicon ← ADD(wordseq, entity, coverage)
14:            end for
15:        end for
16:    end for
17: end procedure
```

3.1.2 Compound Matching algorithm

The matching algorithm is based on a linear search of the entries in the WordSeqLexicon of the source ontology over the standard AML Lexicon built for the target ontologies. All matches are stored on a partial mapping list. A partial mapping corresponds to full string equality between a source word sequence and a target full label. Each partial mapping is assigned a score, called *similarity*, that corresponds to the coverage value of the word sequence weighted by the lexical weight assigned to the target label. This lexical weight is an internal weight given by AML that reflects the relevance of the label (higher for main label, lower for synonyms). This is illustrated in Figure 3.2.

$$similarity = \text{coverage} * \text{weight}$$

For each source class there is a list of partial mappings, each corresponding to a different word sequence. The final compound mapping is built by selecting an appropriate combination of partial mappings. This process is illustrated in Figure 3.3. The selection algorithm begins by sorting the partial mappings in descending order of scores (1). For each label of the source class the algorithm iterates over the partial mappings and adds each of them to an intermediary mapping if (2): the word sequence to which

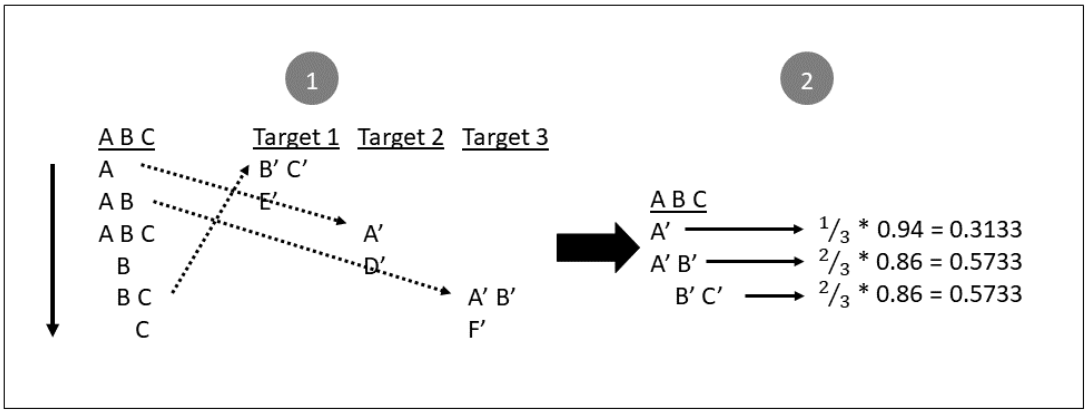

Figure 3.2: Illustration of the matching algorithm.

they were mapped is contained in the current label; the word sequence to which they were mapped was not already covered by a higher score target class. This allows partial overlapping of word sets. By adding a partial mapping to the intermediary mapping, a score is calculated as the average of the similarities of each partial mapping (3). In the special case of having overlapping word sets, the similarity is divided by the number of times the word set appears in the target classes.

$$score = \frac{\sum \frac{\text{similarity}}{\text{frequency of word set}}}{\text{number of words in the target}}$$

In the end, there can be so much as an intermediary mapping per label of a source class. The intermediary mapping with the highest score is chosen, resulting in one compound mapping per source class. Algorithm 2 details this process.

3.1.3 Stemming

As mentioned above, this is a lexical approach, thus making it impossible to find matches between words that have small differences between them, but that do not alter the overall meaning of the concept, such as "ear" and "ears", or "abnormality" and "abnormal". A mapping between these concepts will not be made with this approach if those words are not expressed in the concept's synonyms.

To overcome this issue, a stemming algorithm (Snowball stemmer (Porter & Martin, 2009)) was applied when forming both the Lexicon and WordSeqLexicon, which consists on reducing a word to its stem. For example, by applying the stemmer to "abnormal"

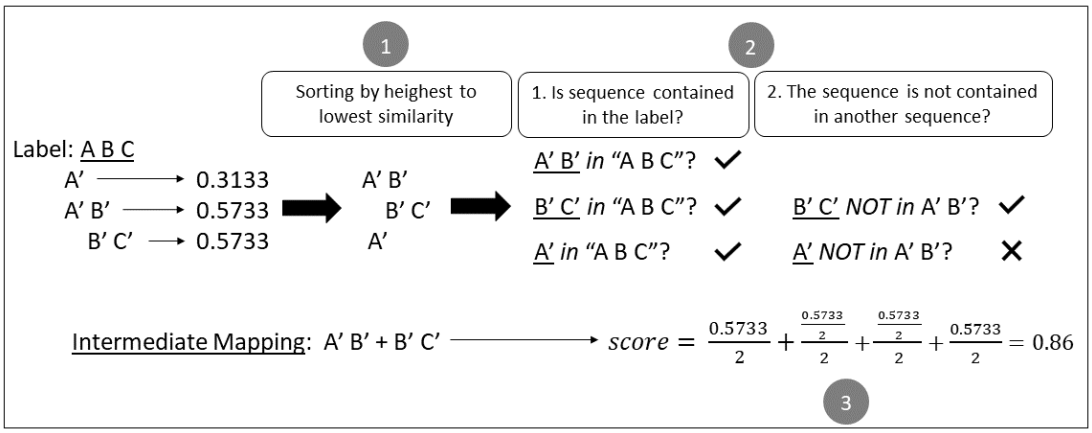

Figure 3.3: Illustration of the filtering algorithm.

and "abnormality", both words would become "abnorm", and thus allowing the mapping to be made by the matcher. Taking this to a full HP concept: "abnormality of hindbrain morphology" (HP:0011282) becomes "abnorm of hindbrain morpholog", and the mapping goes from "hindbrain + shape" (UBERON:0002028 + PATO:0000052) to "abnormal + hindbrain + shape" (PATO:0000460 + UBERON:0002028 + PATO:0000052).

Algorithm 2 WordseqFilterer algorithm

```
1: function FILTER(source, maps)
2:     filteredMaps
3:     sourceBestMaps
4:     sortDescending(maps)
5:     labels ← getAllNames(source)
6:     for label in labels do
7:         for mapping in maps do
8:             target ← target concept in mapping
9:             wordseq ← wordseq in mapping
10:            if wordseq in label then
11:                if wordseq and target not in filteredMaps then
12:                    sourceBestMaps ← ADD(mapping)
13:                end if
14:            end if
15:        end for
16:    end for
17:    bestMapping ← getBest(sourceBestMaps)
18:    filteredMaps ← ADD(mapping)
19:    return filteredMaps
20: end function
```

Chapter 4

Evaluation and Results

The evaluation of the proposed approach was carried out using the Mammalian Phenotype Ontology (MP) (Smith *et al.*, 2004) and Human Phenotype Ontology (HP) (Köhler *et al.*, 2013) as source ontologies. It consisted of two tasks for each source ontology:

(1) Compound alignment using two target ontologies: the Uber Anatomy Ontology (UBERON) (Haendel *et al.*, 2014) and the Phenotypic Quality Ontology (PATO) (Mungall *et al.*, 2010);

(2) Compound alignment using three target ontologies: UBERON, PATO and Gene Ontology (GO) (Ashburner *et al.*, 2000).

The size of the five ontologies is shown in Table 4.1.

Table 4.1: Number of classes in each used ontology.

Ontology	Number of classes
HP	14911
MP	16508
UBERON	15070
PATO	2713
GO	49516

Both source ontologies contain equivalent class axioms that refer to several external ontologies. These were used to produce reference alignments to use in the evaluation.

4.1 Reference alignments generation

The reference alignments were produced following the approach proposed by Pesquita *et al.* (2014), by extracting all the Equivalent Classes Axioms of MP and HP OWL files

using OWL API. For each ontology two references were created:

(1) **UB-PT**: containing mappings derived from equivalent classes axioms that employ classes only from the UBERON and/or PATO ontologies;

(2) **UB-PT-GO**: containing all mappings derived from equivalent classes axioms that employ only the UBERON and/or PATO and GO ontologies.

These references were created as simple text files in TSV format supported by AML. Note that these are just partial alignments, since they only cover 28.6% of the classes in HP and 29.7% in MP for the UB-PT set, and much less for the UB-PT-GO set. The number of mappings in the reference alignments and respective coverage are represented in Table 4.2.

Table 4.2: Number of mappings in each reference alignment and respective coverage.

	Size of reference (mappings)	**Coverage**
HP UB PT	4261	28.6%
HP UB PT GO	463	3.1%
MP UB PT	4896	29.7%
MP UB PT GO	1301	7.9%

4.2 Results

The algorithm was evaluated using eight test cases: (1) HP as source, UBERON and PATO as targets; (2) HP as source, UBERON, PATO and GO as targets; (3) MP as source, UBERON and PATO as targets; (4) MP as source, UBERON, PATO and GO as targets. The remaining four cases are exactly the same, except the results are the ones of the algorithm using the stemmer. The number of mappings produced by each task can be observed in Table 4.3.

Table 4.3: Alignment size for each test case.

		Total
NOT STEMMED	**HP UB PT**	9859
	HP UB PT GO	10134
	MP UB PT	24054
	MP UB PT GO	27378
STEMMED	**HP UB PT**	10665
	HP UB PT GO	10776
	MP UB PT	25748
	MP UB PT GO	28205

AgreementMakerLight allows a threshold as an input to the system that filters the alignment to only present the mappings with the score equal or above the value. The result of applying three thresholds of 0.3, 0.5 and 0.7 is presented in Table 4.4, where it is possible to see that great majority of mappings have less than 0.5 in score.

Table 4.4: Alignment size for each threshold value.

		0.3	0.5	0.7
NOT STEMMED	**HP UB PT**	2063	398	54
	HP UB PT GO	2300	468	60
	MP UB PT	9581	3058	427
	MP UB PT GO	13902	5434	736
STEMMED	**HP UB PT**	2623	579	91
	HP UB PT GO	2868	662	99
	MP UB PT	10582	3504	633
	MP UB PT GO	15011	6028	996

4.2.1 Automatic Evaluation

Given the complexity of tasks, the evaluation of the resulting alignments was performed by classifying each mapping for each source class into one of six orthogonal categories:

Equal The classes in the produced mapping are an exact match to the ones in the reference mapping.

ontology term	small lung
result	**decreased size, lung**
reference	**decreated size, lung**

Contained The classes in the produced mapping are contained in the set of classes in the reference mapping.

ontology term	abnormal submandibular gland physiology
result	**abnormal, submandibular gland**
reference	functionality, **abnormal, submandibular gland**

Containing The classes in the produced mapping contain all classes in the reference mapping.

ontology term	abnormal abdominal wall morphology
result	**abnormal, abdominal wall**, morphology
reference	**abnormal, abdominal wall**

Different The classes in the produced mapping are all different from the classes in the reference mapping.

ontology term	decreased body size
result	size, decreased amount
reference	decreased size, multicellular organism

Overlap The classes in the produced mapping overlap some of the classes in the reference mapping.

ontology term	increased activity of parathyroid
result	increased amount, **parathyroid gland**
reference	**parathyroid gland**, increased rate

Not in results The number of mappings in the results that are not present in the reference alignment.

These categories aim at providing a more fine-grained evaluation of the results. It is

based on a purely lexical view of mapping correctness, which although possibly missing some matches, is easily automated.

Table 4.5 show the number of mappings in each of the six categories. Only a maximum of 12.9% of the mappings found are considered equal to a reference mapping, in the case of the stemmed HP-UB-PT set. The majority of mappings falls into either the $Different$ or the *Equal* categories. Most of the considered correct mappings (Equal, Contained and Containing) fall into the category of *Equals*, except in the case of Not Stemmed HP UB PT and HP UB PT GO. The results of these sets significantly improve when the Stemmer algorithm is applied during the generation of the lexicons. Similarly there are good improvements when applying the Stemmer to all the other cases.

Table 4.5: Number of mappings in each of the categories.

		Equals	Contained	Contains	Different	Overlap	Not in results	Total	Reference
NS	HP UB PT	754	1023	55	2191	1234	234	9859	4261
	HP UB PT GO	43	149	6	208	145	53	10134	463
	MP UB PT	2641	736	205	1217	1140	93	24054	4896
	MP UB PT GO	554	87	41	598	468	17	27378	1301
S	HP UB PT	1375	441	132	2177	1572	132	10665	4261
	HP UB PT GO	115	65	11	277	177	41	10776	463
	MP UB PT	2813	585	222	1195	1177	77	25748	4896
	MP UB PT GO	567	70	30	615	485	15	28205	1301

4.2.1.1 Performance metrics

Using the first three categories, two versions of true positive mappings were computed in order to account for partial matching. These two versions of true positive mappings, represent partially correct mappings that are still considered useful. In compound matching, the complexity and difficulty of the task is such, that an interactive alignment scenario where potentially correct mappings are shown to a user for editing or validating is highly likely. As such, evaluating the approach considering partially correct mappings more accurately measures the usefulness of the proposed matching approach.

(1) *permissive TP*, where a mapping is considered positive if it is equal to or contained by a reference mapping;

$$\text{permissive TP} = Equal + Contained$$

(2) *fuzzy TP*, where a mapping is considered positive if it has at least one matched class contained in the reference mapping:

$$\text{fuzzy TP} = Equal + Contained + Containing + Overlap$$

Using the classical true positive and the two true positive variants, *precision*, *recall* and *f-measure* scores were calculated, shown in Figures 4.1, 4.2 and 4.3, respectively, for both stemmed and not stemmed sets:

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

$$\text{f-measure} = 2 * \frac{precision * recall}{precision + recall}$$

Using the classical version of true positive evaluation, i.e., considering only the Equal mappings, the metrics are significantly lower than when using the permissive true positives. Comparing the latter to the fuzzy evaluation, the metrics present lower values also. Using the more permissive evaluation, all three metrics increase. In general, when the stemming algorithm is applied, the performance increases.

In general, precision is higher for HP sets than for MP sets, but when GO ontology is present, this metric is more or less similar between the HP and MP sets. The fuzzy precision was calculated taking into account not only the mappings in the Equal category, but also in the *Contained*, *Contains* and *Overlap* categories, and improved especially for HP tasks.

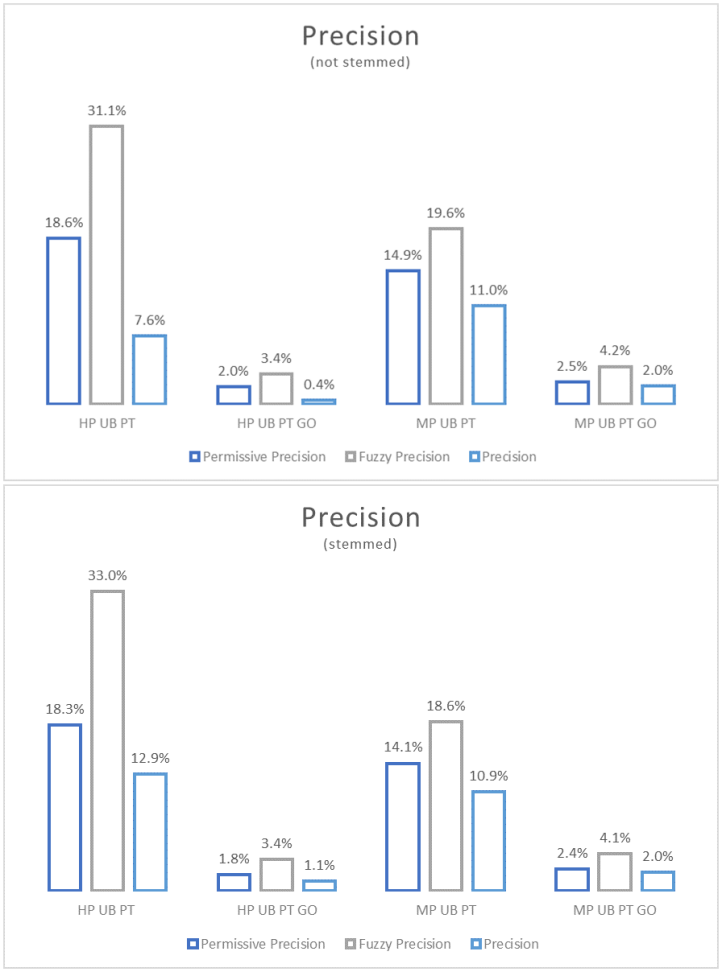

Figure 4.1: Permissive, fuzzy and normal precision for not stemmed and stemmed sets.

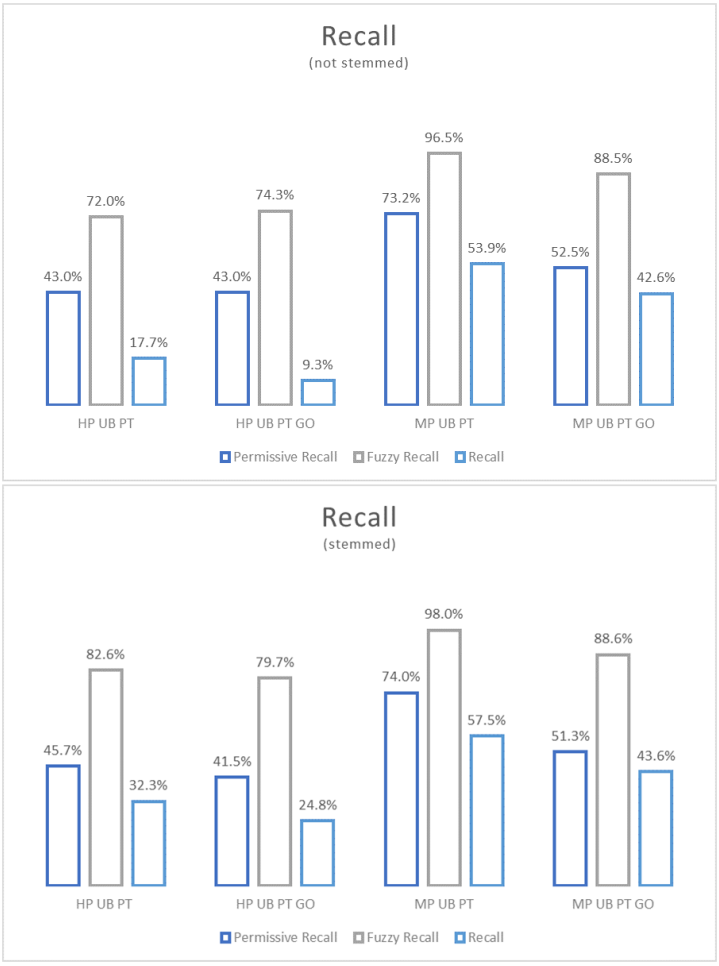

Figure 4.2: Permissive, fuzzy and normal recall for not stemmed and stemmed sets.

Regarding the recall, the general situation is the same as the results improve from the normal recall to the permissive recall, and from this to the fuzzy recall. There are improvements from not stemmed sets to the stemmed sets in all cases, except in permissive recall when GO is introduced, where this value slightly lowers. The lowest recall is normal recall of 9.5% for not stemmed HP-UB-PT-GO and the highest is fuzzy recall of 98% for stemmed MP-UB-PT.

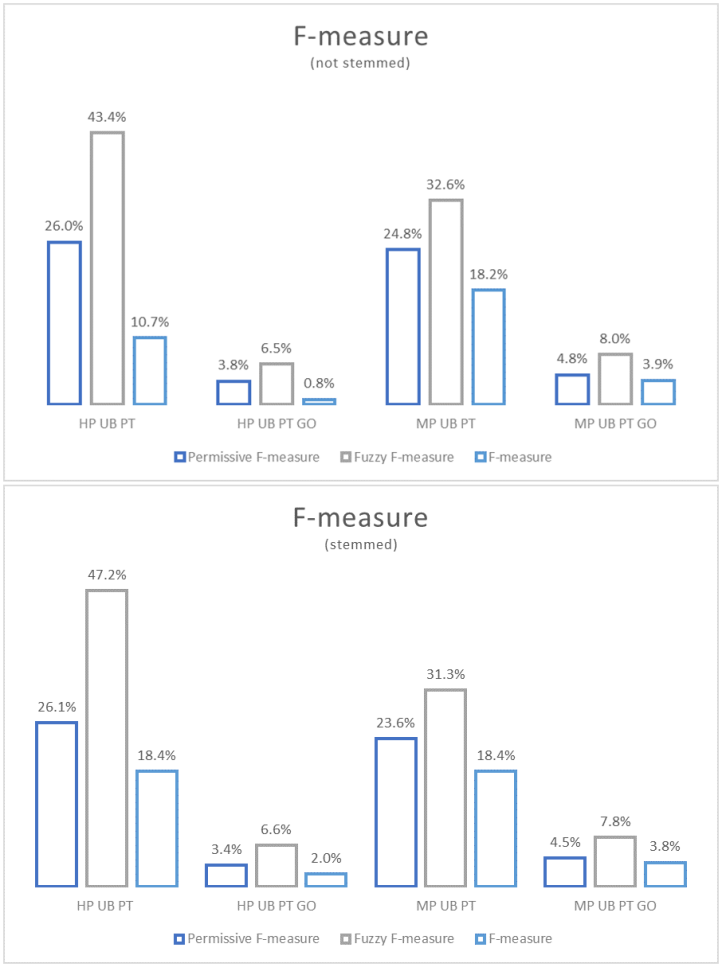

Figure 4.3: Permissive, fuzzy and normal f-measure for not stemmed and stemmed sets.

The f-measure was calculated as an average of the previous two statistics. For the normal metric results vary between 0.8% for not stemmed HP-UB-PT-GO and 18.4% for Stemmed HP-UB-PT. The permissive f-measure shows values between 3.4% for stemmed HP-UB-PT-GO and 26.1% for stemmed HP-UB-PT and the fuzzy f-measure has its minimum at 6.5% for both stemmed and not stemmed HP-UB-PT-GO and its maximum at 47.2% for stemmed HP-UB-PT.

4.2.2 Top mappings analysis

During the algorithm processing, at a given step, several different combinations of mappings are produced and sorted by score, from the highest to the lowest. These were

taken to perform an analysis and see in which cases the correct mapping (i.e. the mapping in the reference) was present on the first, three or five first mappings. This analysis was performed in order to simulate an interactive matching process, where the users are shown the best mappings for a given source class and are able to select the correct one.

The results are displayed in Table 4.6, where TOP 1 corresponds to the Equals category from before. The graphics show that indeed there are correct mappings that are formed and are scored from second to fifth and do not reach the final results.

Table 4.6: Number of mappings in Top 1, Top 3 and Top 5.

		TOP1	TOP3	TOP5	Total	Reference
NS	**HP UB PT**	754	847	901	9859	4261
	HP UB PT GO	43	50	50	10134	463
	MP UB PT	2641	2725	2905	24064	4896
	MP UB PT GO	554	556	576	27378	1301
S	**HP UB PT**	1375	1580	1652	10665	4261
	HP UB PT GO	115	129	131	10776	463
	MP UB PT	2813	2853	3053	25748	4896
	MP UB PT GO	567	564	585	28205	1301

As expected, more correct mappings are found when increasing the range from 1 to 3 and to 5. However, the improvement is still modest.

From the previous results, the precision, recall and f-measure metrics were calculated for the TOP 3 and TOP 5 mappings and can be seen in Figures 4.4, 4.5 and 4.6, respectively. The pattern is the same in the three cases, showing an increase from TOP1 to TOP 5. Adding GO to the test sets, the precision and f-measure are significantly lower in all sets. In the same situation, recall also presents lower values but not as low. In general, when the stemmer algorithm is applied, the results improve for the three metrics.

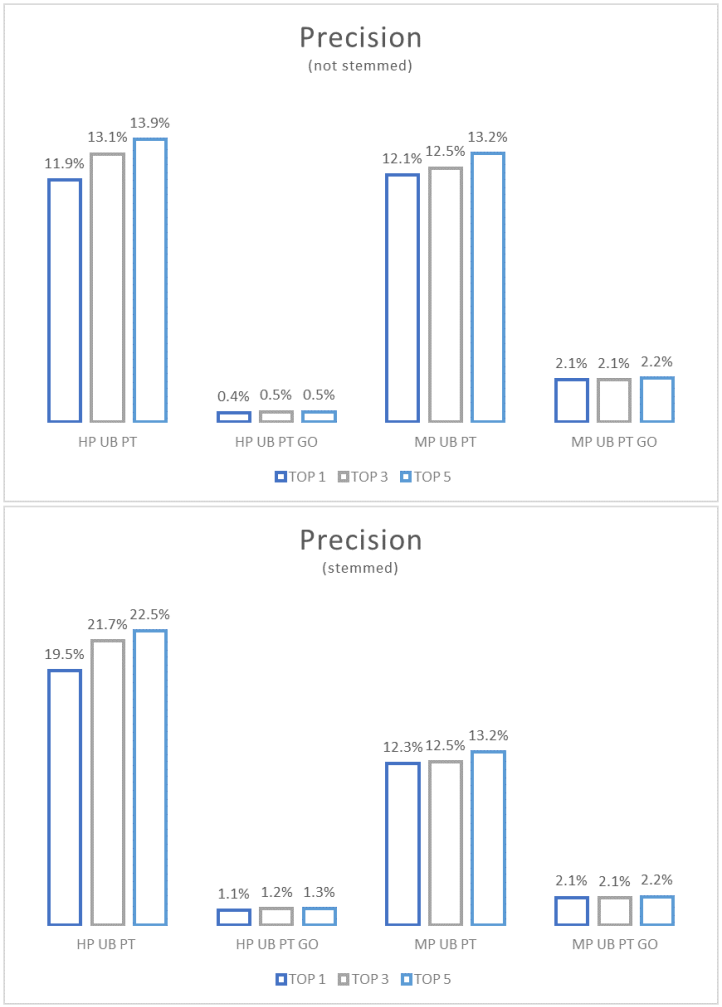

Figure 4.4: Precision of the TOP analysis.

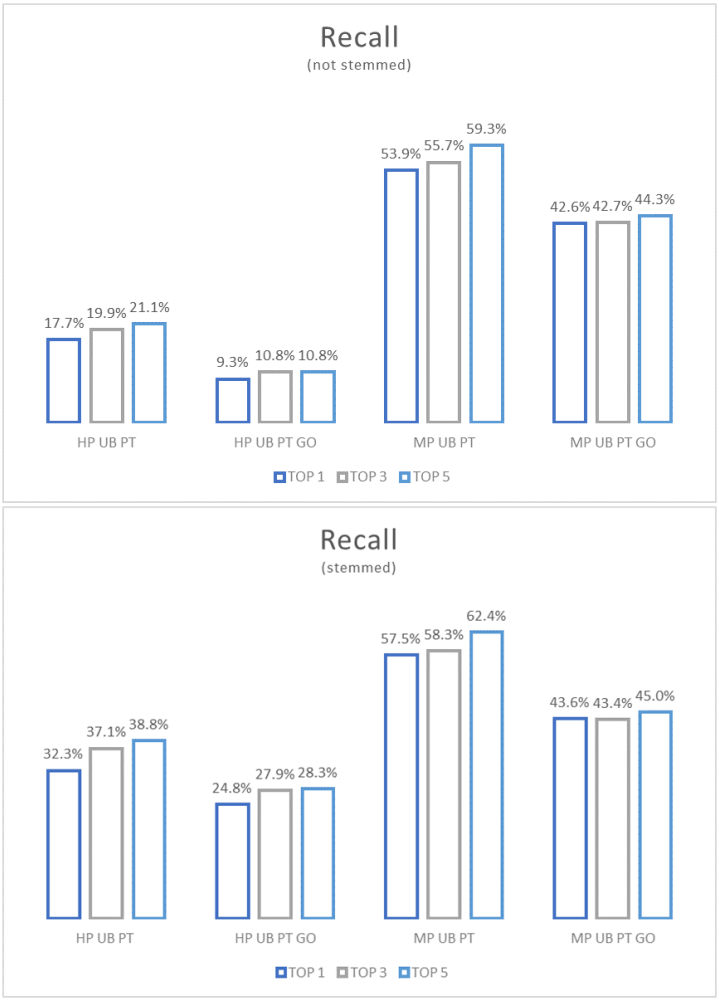

Figure 4.5: Recall of the TOP analysis.

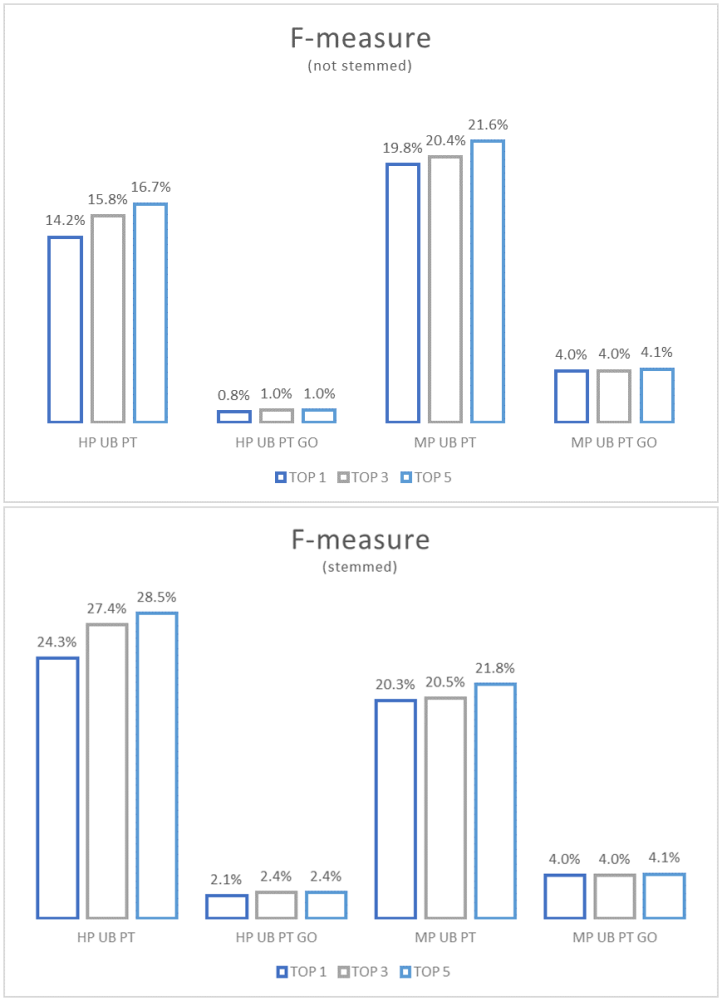

Figure 4.6: F-measure of the TOP analysis.

4.2.3 Running time analysis

When dealing with big ontologies and with algorithms that perform linear search, it can be expected the process takes some time. In this sense, a running time assessment was performed for three phases: *All*, the running time of all the process; *Match+filter*, corresponding to the time it takes to run the matching and filtering algorithms; *Filter*, the time it takes to filter the final mappings. These results can be observed in Figure 4.7 for both stemmed and not stemmed sets. The analysis was performed in a machine with the following characteristics: operating system Windows 10 64bits, processor Intel® Core™ i5-6200U CPU @ 2.40GHz and 8.00GB of RAM.

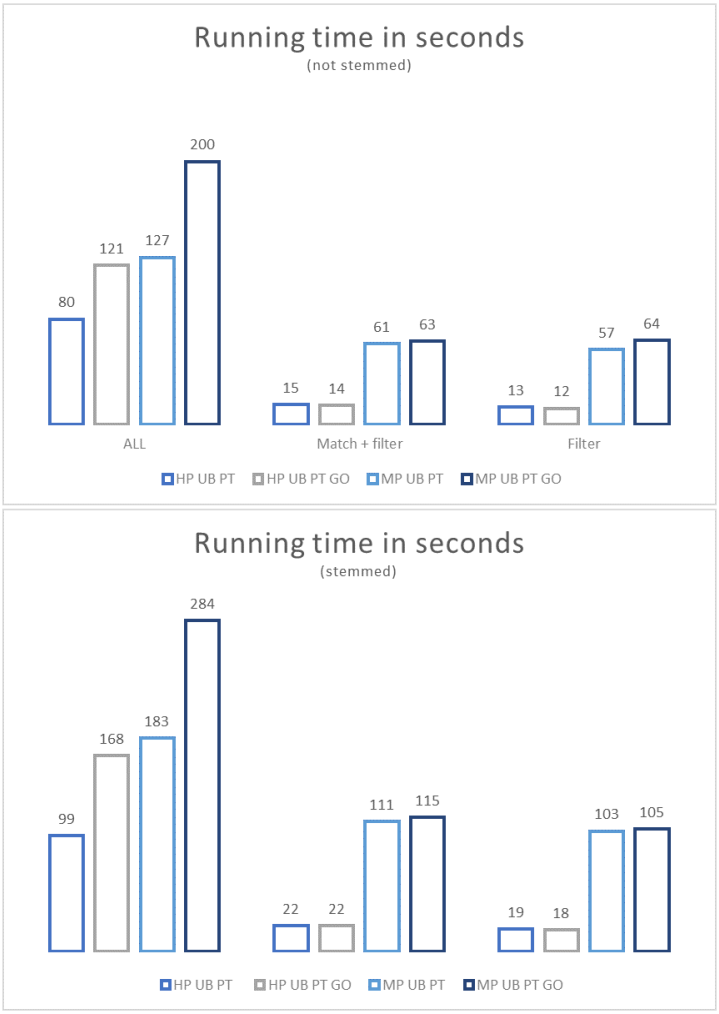

Figure 4.7: Running time of the three main phases of the AML algorithm.

The running time is always higher for tasks in which the source ontology is MP. However, during matching and filtering, the time does not vary much between tasks with same source ontology. When GO ontology is involved, the overall running time increases considerably.

Chapter 5

Discussion

Performing a full compound mapping can be a rather complex task, given that it involves multiple classes from multiple ontologies which need to be matched, and the best combination between the various partial mappings needs to be found. In this work, the generated alignments were compared in an automatic evaluation against reference alignments and it was considered that even if a mapping is not fully correct, it may still be useful. Partial matching has been found to be useful in the biomedical context (Dhombres & Bodenreider, 2016). Given the complexity of performing compound matching, a scenario where it is performed in a semi-automated version with human input is highly likely. As such, we have followed an evaluation based on partially correct mappings, under the assumption that these could then be shown to a user for a final decision.

However, automated evaluation is further hindered by the fact that the reference alignments used for evaluation are not complete. This makes it impossible to evaluate the mappings that were created for a source class not contained in the reference. This is reflected in the performance metrics that were computed, with precision always being lower than recall. Even when using fuzzy performance metrics, values for precision in the best task fall short of 40% whereas recall hits 98%. Using the proposed evaluation metrics allows a better understanding of the potential usefulness of the proposed approach, with precision tripling in some cases (from classical to fuzzy definitions), and recall nearly doubling (or more) in all cases.

All performance metrics significantly improve when the Stemmer algorithm is applied during the generation of the lexicons, which suggests that especially in HP there

were a lot of terms not being matched that had small differences between the source and the target, such as plurals or spelling variants.

While using these relaxed metrics, illustrates the usefulness of the approach in an interactive scenario where a partially correct mapping is shown to the user for editing, another possibility is showing the user multiple options of target class expressions that map to a single source class for the user to select the correct one. The TOP 3 and 5 analysis was performed in order to test this scenario. The results show that indeed the top scoring mapping is not always the "right" mapping, but in a few cases it can be found in the top scoring mappings set.

The MP ontology performs better in general than HP, this is likely due to the fact that MP uses vocabulary that is more similar to the one employed in PATO and UBERON. In fact, while HP is restricted to human anatomy, MP covers all mammalian anatomy and UBERON is species-agnostic.

When GO is present as a target ontology, the number of mappings increases in the alignment. The precision and f-measure are much lower when GO is present but the recall is practically the same. This is unsurprising given that adding a third ontology increases the search space and the probability of creating a mapping increases. However, given the incompleteness of the reference alignments, a higher number of mappings also results in lower precision. Moreover, when creating the reference alignments with GO, only equivalent classes that comprised GO were taken into account, producing much smaller reference alignments.

One of the biggest problems faced when matching ontologies, and in particular Biomedical ontologies, is the running time when large ontologies are used. Here, the running time is short comparing to the approach developed by Oliveira (2015), where, when using an Intel® Core™i7-2600 CPU 3.40GHz and 16GB of RAM and only two target ontologies, the running time was of over 15h when the threshold was set to 0.1, even when using large ontologies such as GO, and in this approach, the longest running time was of less than five minutes. It is, in comparison, an algorithm of simpler use, given that it's not required to take into account the order of the ontologies to use as targets, neither does it restrict the number of terms matched per target ontology, as the algorithm will first match all the available terms in all the ontologies and then create the mappings, allowing for more complete alignments. For example for the set MP-UB-PT this algorithm got 24054 mappings against 1413 in Oliveira & Pesquita (2018), where

the matching is restricted. On the other hand, this is a purely lexical approach and by keeping not only the labels and the synonyms but also their combinations in dedicated structures, it is more memory intensive.

Chapter 6

Conclusions and Future Work

This MSc project proposed, developed and evaluated a novel compound matching algorithm, able to compose mappings between a source class and a class expression relating multiple classes from multiple target ontologies. It addresses the limitations of previous approaches that only considered two target classes from two target ontologies. The algorithm is based on the efficient lexical matching approaches in AgreementMakerLight, and was evaluated on a set of partial compound alignments.

When using classical performance metrics, the obtained results were poor, with a top f-measure of 24%. The difficulties in performing complex alignments are well known, and a recent evaluation of complex matching approaches revealed that all techniques produced f-measures below 20% (Thiéblin *et al.*, 2018b). This difficulty is easily translated into compound alignments as well, since they share many of the same challenges. The somewhat low results obtained can also, at least partially, be explained by the fact that the built reference alignments can only be considered partial references. Not only do they cover less than 30% of the ontologies, it has been previously shown that between 60 and 90% of ternary compound mappings found are not captured in the equivalent class axioms (Oliveira & Pesquita, 2018). There are recent efforts in building reference alignments for complex matching (Thiéblin *et al.*, 2018b) that highlight the growing interest in promoting complex matching. However, building these references is a highly time-consuming task.

However, an evaluation that focuses on measuring the usefulness of the proposed approach for an interactive alignment scenario revealed more promising results with recall values between 80 and 98%.

6. CONCLUSIONS AND FUTURE WORK

There are several future work endeavors in this area. For instance, improving the lexicon algorithm to allow for more efficient data structures, integrating the approach in the graphical user interface of AML to support alignment with human interaction, and producing a scoring function for mappings that better reflects their similarity to human users.

Finally, the current version of the algorithm is focused on finding the classes involved in the equivalent class axioms, however such axioms also contain several property restrictions, and thus to be able to fully reproduce the axiom, both the classes and the properties involved would need to be mapped. This represents another layer of complexity, since properties present specific challenges for ontology matching algorithms (Cheatham & Hitzler, 2014).

Compound ontology matching has been proposed as a technique to enrich ontologies with equivalence class axioms (Oliveira & Pesquita, 2018). It could also be adapted to the integration of multidimensional semantic spaces (Berlanga *et al.*, 2012), or to enrich the Linked Open Vocabularies (Vandenbussche *et al.*, 2017) with more complex mappings. The impact of compound in the field of biomedical ontologies can be considerable, regarding the heterogeneity of biomedical data and the number of existing biomedical ontologies.

References

An, Y., Borgida, A. & Mylopoulos, J. (2005a). Constructing complex semantic mappings between xml data and ontologies. In *International Semantic Web Conference*, 6–20, Springer. 13

An, Y., Borgida, A. & Mylopoulos, J. (2005b). Inferring complex semantic mappings between relational tables and ontologies from simple correspondences. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, 1152–1169, Springer. 13

Arnold, P. (2013). Semantic enrichment of ontology mappings: Detecting relation types and complex correspondences. *Grundlagen von Datenbanken*, **1020**, 34–39. 13

Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T. *et al.* (2000). Gene Ontology: tool for the unification of biology. *Nature genetics*, **25**, 25–29. 6, 21

Berlanga, R., Jiménez-Ruiz, E. & Nebot, V. (2012). Exploring and linking biomedical resources through multidimensional semantic spaces. *BMC bioinformatics*, **13**, 1. 40

Berners-Lee, T., Hendler, J. & Lassila, O. (2001). The semantic web. *Scientific american*, **284**, 34–43. 2

Bodenreider, O., Mitchell, J.A. & McCray, A.T. (2005). Biomedical ontologies. In *Pacific Symposium on Biocomputing*, 76–78. 6

Chandrasekaran, B., Josephson, J. & Benjamins, V.R. (1999). What are ontologies, and why do we need them? *Intelligent Systems and their Applications, IEEE*, **14**, 20 – 26. 5

REFERENCES

Cheatham, M. & Hitzler, P. (2014). The properties of property alignment. In *OM*, 13–24. 40

Cruz, I.F., Antonelli, F.P. & Stroe, C. (2009). Agreementmaker: efficient matching for large real-world schemas and ontologies. *Proceedings of the VLDB Endowment*, **2**, 1586–1589. 11

Dhombres, F. & Bodenreider, O. (2016). Interoperability between phenotypes in research and healthcare terminologies—Investigating partial mappings between HPO and SNOMED CT. *Journal of Biomedical Semantics*, **7**, 3. 35

Djeddi, W.E. & Khadir, M.T. (2010). Xmap: a novel structural approach for alignment of owl-full ontologies. In *2010 International Conference on Machine and Web Intelligence*, 368–373, IEEE. 12

Doan, A., Madhavan, J., Dhamankar, R., Domingos, P. & Halevy, A. (2003). Learning to match ontologies on the semantic web. *The VLDB journal*, **12**, 303–319. 13

Dou, D., Qin, H. & Lependu, P. (2010). Ontograte: Towards automatic integration for relational databases and the semantic web through an ontology-based framework. *International Journal of Semantic Computing*, **4**, 123–151. 13

Duchateau, F., Coletta, R., Bellahsene, Z. & Miller, R.J. (2009). Yam: a schema matcher factory. In *Proceedings of the 18th ACM conference on Information and knowledge management*, 2079–2080, ACM. 12

Euzenat, J., Shvaiko, P. *et al.* (2007). *Ontology matching*, vol. 18. Springer, Berlin, Heidelberg. 6, 9

Faria, D., Pesquita, C., Santos, E., Palmonari, M., Cruz, I.F. & Couto, F.M. (2013). The agreementmakerlight ontology matching system. In *On the Move to Meaningful Internet Systems: OTM 2013 Conferences*, 527–541, Springer, Berlin, Heidelberg. 3, 11, 15

Ferreira, J.D., Pesquita, C., Couto, F.M. & Silva, M.J. (2012). Bringing epidemiology into the semantic web. In *ICBO*. 2

Golbreich, C., Horridge, M., Horrocks, I., Motik, B. & Shearer, R. (2007). Obo and owl: Leveraging semantic web technologies for the life sciences. In *The Semantic Web*, 169–182, Springer. 8

Gruber, T.R. (1993). A translation approach to portable ontology specifications. *Knowledge acquisition*, **5**, 199–220. 5

Haendel, M.A., Balhoff, J.P., Bastian, F.B., Blackburn, D.C., Blake, J.A., Bradford, Y., Comte, A., Dahdul, W.M., Dececchi, T.A., Druzinsky, R.E., Hayamizu, T.F., Ibrahim, N., Lewis, S.E., Mabee, P.M., Niknejad, A., Robinson-Rechavi, M., Sereno, P.C. & Mungall, C.J. (2014). Unification of multi-species vertebrate anatomy ontologies for comparative biology in Uberon. *Journal of Biomedical Semantics*, **5**, 21. 21

Hoehndorf, R., Schofield, P.N. & Gkoutos, G.V. (2015). The role of ontologies in biological and biomedical research: A functional perspective. *Briefings in Bioinformatics*, **16**, 1069–1080. 7

Horridge, M., Jupp, S., Moulton, G., Rector, A., Stevens, R. & Wroe, C. (2009). A practical guide to building owl ontologies using protégé 4 and co-ode tools edition1. 2. *The university of Manchester*, **107**. 8

Hu, W. & Qu, Y. (2006). Block matching for ontologies. In *The Semantic Web - ISWC 2006*, 300–313, Springer Berlin Heidelberg. 13

Hu, W., Chen, J., Zhang, H. & Qu, Y. (2011a). Learning complex mappings between ontologies. In *Joint International Semantic Technology Conference*, 350–357, Springer. 13

Hu, W., Chen, J., Zhang, H. & Qu, Y. (2011b). Learning complex mappings between ontologies. In *JIST*. 13

Jiang, S., Lowd, D., Kafle, S. & Dou, D. (2016). Ontology matching with knowledge rules. In *Transactions on Large-Scale Data-and Knowledge-Centered Systems XXVIII*, 75–95, Springer. 13

REFERENCES

Jiménez-Ruiz, E. & Cuenca Grau, B. (2011). Logmap: logic-based and scalable ontology matching. In *The Semantic Web–International Semantic Web Conference (ISWC)*, 273–288, Springer Berlin/Heidelberg. 11

Jiménez-Ruiz, E., Grau, B.C. & Cross, V.V. (2016). Logmap family participation in the oaei 2016. In *OM@ISWC*. 11

Köhler, S., Doelken, S.C., Mungall, C.J., Bauer, S., Firth, H.V., Bailleul-Forestier, I., Black, G.C., Brown, D.L., Brudno, M., Campbell, J. *et al.* (2013). The Human Phenotype Ontology project: linking molecular biology and disease through phenotype data. *Nucleic acids research*, gkt1026. 21

Maedche, A., Motik, B., Silva, N. & Volz, R. (2002). Mafra—a mapping framework for distributed ontologies in the semantic web. In *Workshop on knowledge Transformation for the Semantic Web, Lyon, France, ECAI*, 60–68. 12

McGuinness, D.L., Van Harmelen, F. *et al.* (2004). Owl web ontology language overview. *W3C recommendation*, **10**, 2004. 8

Mungall, C.J., Gkoutos, G.V., Smith, C.L., Haendel, M.A., Lewis, S.E. & Ashburner, M. (2010). Integrating phenotype ontologies across multiple species. *Genome biology*, **11**, R2. 21

Musen, M.A., Middleton, B. & Greenes, R.A. (2014). Clinical decision-support systems. In *Biomedical informatics*, 643–674, Springer. 8

Ngo, D.H. & Bellahsene, Z. (2012). Yam++:(not) yet another matcher for ontology matching task. In *BDA: Bases de Données Avancées*. 12

Nunes, B.P., Mera, A., Casanova, M.A., Fetahu, B., Leme, L.A.P.P. & Dietze, S. (2013). Complex matching of rdf datatype properties. In *International Conference on Database and Expert Systems Applications*, 195–208, Springer. 13

Oliveira, D. & Pesquita, C. (2015). Compound matching of biomedical ontologies. *In International Conference on Biomedical Ontology (ICBO)*. 11

Oliveira, D. & Pesquita, C. (2018). Improving the interoperability of biomedical ontologies with compound alignments. *Journal of biomedical semantics*, **9**, 1. 2, 13, 36, 39, 40

Oliveira, D.P.d.S. (2015). *Compound matching of biomedical ontologies*. Master's thesis, University of Lisbon. 36

Ontotext, *What are Ontologies?* `https://www.ontotext.com/knowledgehub/fundamentals/what-are-ontologies/` [Online; accessed 2019-04-09]. 6

Otero-cerdeira, L., Rodríguez-martínez, F.J. & Gómez-rodríguez, A. (2015). Expert Systems with Applications Ontology matching : A literature review. *EXPERT SYSTEMS WITH APPLICATIONS*, **42**, 949–971. 9, 10

Parundekar, R., Knoblock, C.A. & Ambite, J.L. (2010). Linking and building ontologies of linked data. In *International Semantic Web Conference*, 598–614, Springer. 12

Parundekar, R., Knoblock, C.A. & Ambite, J.L. (2012). Discovering concept coverings in ontologies of linked data sources. In *International Semantic Web Conference*, 427–443, Springer. 13

Pesquita, C., Cheatham, M., Faria, D., Barros, J., Santos, E. & Couto, F.M. (2014). Building reference alignments for compound matching of multiple ontologies using obo cross-products. In *Ontology Matching Workshop at ISWC 2014*. 9, 13, 21

Porter & Martin (2009). Snowball: A language for stemming algorithms, 2001. *URL http://snowball.tartarus.org/texts/introduction.html (visited April 2019)*. 18

Qin, H., Dou, D. & LePendu, P. (2007). Discovering executable semantic mappings between ontologies. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, 832–849, Springer. 13

Ritze, D., Meilicke, C., Šváb Zamazal, O. & Stuckenschmidt, H. (2009). A pattern-based ontology matching approach for detecting complex correspondences. In *Proceedings of the 4th International Conference on Ontology Matching*, vol. 551 of *OM'09*, 25–36, CEUR-WS.org, Aachen, Germany, Germany. 2, 12, 13

Ritze, D., Völker, J., Meilicke, C. & Šváb Zamazal, O. (2010). Linguistic analysis for complex ontology matching. In *Proceedings of the 5th International Conference*

on Ontology Matching, vol. 689 of *OM10*, 1–12, CEUR-WS.org, Aachen, Germany, Germany. 12, 13

Robinson, P. & Bauer, S. (2011). *Introduction to Bio-ontologies*. 6

Scharffe, F. (2009). *Correspondence patterns representation*. Ph.D. thesis, PhD thesis, University of Innsbruck. 13

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**, 1251–1255. 1

Smith, C.L., Goldsmith, C.A.W. & Eppig, J.T. (2004). The Mammalian Phenotype Ontology as a tool for annotating, analyzing and comparing phenotypic information. *Genome biology*, **6**, R7. 21

Thiéblin, É., Haemmerlé, O., Hernandez, N. & Trojahn, C. (2018a). Survey on complex ontology matching. *Semantic Web Journal*. 9, 13

Thiéblin, É., Haemmerlé, O., Hernandez, N. & Trojahn, C. (2018b). Task-oriented complex ontology alignment: Two alignment evaluation sets. In *European Semantic Web Conference*, 655–670, Springer. 12, 39

Vandenbussche, P.Y., Atemezing, G.A., Poveda-Villalón, M. & Vatant, B. (2017). Linked open vocabularies (lov): a gateway to reusable semantic vocabularies on the web. *Semantic Web*, **8**, 437–452. 40

Walshe, B., Brennan, R. & O'Sullivan, D. (2016). Bayes-recce: A bayesian model for detecting restriction class correspondences in linked open data knowledge bases. *International Journal on Semantic Web and Information Systems (IJSWIS)*, **12**, 25–52. 12, 13

Whetzel, P.L., Noy, N.F., Shah, N.H., Alexander, P.R., Nyulas, C., Tudorache, T. & Musen, M.A. (2011). Bioportal: enhanced functionality via new web services from the national center for biomedical ontology to access and use ontologies in software applications. *Nucleic acids research*, **39**, W541–W545. 8