UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

Compound Matching of Biomedical Ontologies

Mestrado em Bioinformática e Biologia Computacional
Especialização em Bioinformática

Daniela Patrícia dos Santos Oliveira

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

2015

Resumo

As ontologias biomédicas são particularmente bem sucedidas na uniformização do domínio das ciências da vida. Devido à sua recente expansão, a integração de todo o conhecimento que contêm tornou-se uma tarefa extenuante. Desta forma, foram desenvolvidos sistemas de alinhamento de ontologias para lidar com o problema, que alinham uma ontologia com outra e encontram classes que correspondem entre as duas. Contudo, novos desafios para estes sistemas estão a começar a aparecer, já que existem ontologias biomédicas que contêm relações complexas e os sistemas têm dificuldade em encontrá-las. Produzir alinhamentos compostos, ou seja, que alinham mais de duas ontologias simultaneamente, pode ser útil para o desenvolvimento de uma próxima geração de tecnologias semânticas.

Desta forma, esta dissertação avança o campo de alinhamento de ontologias biomédicas com o desenvolvimento de novos algoritmos que produzem correspondências compostas entre três ontologias diferentes, uma fonte e dois alvos. O algoritmo é baseado em dois passos de comparação léxica. Num primeiro é feito o alinhamento parcial da ontologia fonte com um primeiro alvo, e no segundo apenas as palavras não mapeadas das classes fonte alinhadas são comparadas com as palavras das classes do segundo alvo. O alinhamento composto assim gerado é sujeito a um passo de seleção para encontrar a melhor correspondência possível para cada classe da fonte.

Os alinhamentos resultantes foram avaliados contra seis alinhamentos de referência automaticamente inferidos a partir de definições lógicas de ontologias biomédicas da *OBO Foundry*, mas também foram manualmente verificados. Os resultados preliminares, usando a avaliação automática, apresentam f-measure baixa, com uma precisão mais

alta, que flutua entre os 62.9 e os 11.7% e sensibilidade máxima de 60.7%. Contudo, a análise manual demonstra que, apesar do baixo desempenho contra as referências automáticas, o algoritmo estava a encontrar maioritariamente mapeamentos corretos, com uma pequena percentagem de mapeamentos incorretos.

Assim, esta descoberta inspirou a investigação da possível aplicação do algoritmo na expansão e manutenção das definições lógicas. O algoritmo também foi bem sucedido no alinhamento de conjuntos ternários de ontologias do domínio das plantas.

Palavras Chave: ontologias biomédicas, emparelhamento de ontologias, alinhamento de ontologias, alinhamento composto de ontologias, definições lógicas

Abstract

Biomedical ontologies are particularly successful in the uniformization of the life sciences domain. Due to their recent expansion it became a strenuous task to integrate all the knowledge they encompass. So, ontology matching systems were developed to deal with the problem by aligning one ontology to another and finding matching classes. However, there are still some challenges which are not addressed by the current systems, since there are ontologies which cover complex relations and they struggle to find them. Therefore, I argue that producing "compound" alignments, which match more than two ontologies simultaneously, could be potentially useful to support a next generation of semantic technologies.

This thesis advances the field of ontology matching with the development of novel algorithms that produce compound matches between three different ontologies. The overall steps of the algorithm involve matching a source ontology to a first target and, from the resulting alignment, the source classes not mapped are removed and the words already matched are ignored in the second matching step. This second step aligns those remaining words to the third ontology and returns a compound alignment, which is subjected to a selection step to find the best possible match for each source class.

The resulting alignments were evaluated against six reference alignments automatically inferred from logical definition of biomedical ontologies, but they were also manually to verified. Preliminary results using the automatic evaluation approach present low f-measure, with a higher precision, which fluctuates between 62.9 and 11.7% and the higher recall is 60.7%. However, the manual analysis showed that despite the low performance against the automatic references, the

algorithm was obtaining mostly correct mappings, with a very low percentage of incorrect mappings.

Therefore, this finding led me to think that the reference alignments can be expanded and so, one of the possible applications of this algorithm could be to help experts add and maintain the logical definitions present in the OBO Foundry. The algorithm was also successful in its application to align several ternary sets of plant related ontologies.

Keywords: biomedical ontologies, ontology matching, ontology alignment, compound ontology matching, logical definitions

Resumo Alargado

O desenvolvimento e proliferação de novas técnicas de Bioinformática, como as novas técnicas de sequenciação automática de DNA, e a crescente adoção e uso de sistemas de saúde informáticos, aumentou consideravelmente a quantidade de dados geridos pelas ciências biomédicas. Consequentemente, é cada vez mais relevante para os investigadores terem uma forma prática e eficaz de aceder aos dados provenientes de diversas fontes. Nesta área, as ontologias biomédicas têm sido bem sucedidas a integrar o conhecimento, uma vez que permitem representá-lo de uma forma relativamente uniforme em diferentes domínios. Contudo, o desenvolvimento não coordenado destas ontologias levou à criação de ontologias diferentes para domínios iguais ou relacionados, dificultando a partilha e integração de conhecimento.

Atualmente, já foram desenvolvidos vários sistemas de alinhamento de ontologias, que estão a ser continuamente melhorados, e têm como objetivo mapear classes entre duas ontologias diferentes. A tarefa de alinhamento de ontologias biomédicas é árdua uma vez que, devido elevada dimensão e complexidade das ontologias, pode tornar-se complicado obter alinhamentos satisfatórios em tempo útil. Os esforços na área do alinhamento de ontologias biomédicas têm sido especialmente direcionados para a descoberta de mapeamentos entre apenas duas ontologias, o que, em ontologias de domínios tão diversos e complexos como o da biomedicina, pode não encontrar todo o tipo de relações que existe dentro das ontologias.

Neste sentido, o objetivo central desta tese é desenvolver métodos automáticos para a descoberta de equivalências entre classes de ontologias através da criação de um algoritmo que seja capaz de mapear classes, não de duas, mas de três ontologias diferentes, formando mapeamentos "Compostos", que encontram equivalências de classes de

uma ontologia A, com classes que se intersectam de duas ontologias B e C.

Inicialmente foi feita uma análise de alinhamentos entre ontologias duas a duas, para verificar se haveria algum tipo de padrão que pudesse ser proveitoso para o desenvolvimento do algoritmo. Os padrões encontrados permitiram desenvolver o conceito inicial do algoritmo central, uma vez que vários termos das classes tinham correspondência exata dentro dos termos de outra ontologia, mas continham pequenas diferenças, como a adição ou variação de palavras. Este tipo de padrão permitiu perceber que uma abordagem de alinhamento palavra-a-palavra devesse ser a mais indicada e levou a que certas experiências fossem testadas de forma a melhorar o algoritmo base.

Desta forma, foi adotada uma estratégia de alinhamento palavra-a-palavra, mas foi necessária uma especial atenção para o potencial aumento de espaço de procura que este tipo de algoritmo pode ter e, com esse intuito, desenvolvi o algoritmo de forma a que o alinhamento fosse obtido em dois passos principais que permitem reduzir o espaço de procura. Resumidamente, o primeiro passo alinha a ontologia de origem (O_s) com a primeira ontologia alvo (O_{t1}). O alinhamento resultante contém mapeamentos de classes das duas ontologias que partilham palavras, com um valor de semelhança entre elas. Esta semelhança é uma fração ponderada pelo *Evidence Content* (EC), que exprime a frequência com que um termo aparece no léxico das ontologias. Assim, a soma do EC dos termos partilhados pelas duas classes é dividida pelo peso total dos termos da classe O_{t1}. Para o segundo passo são removidas todas as classes da O_s que não foram mapeadas com nenhum termo de O_{t1} e são também removidos os termos dos mapeamentos que já têm correspondência com termos de O_s. Com este processo de subtração é possível reduzir o espaço de procura e, desta forma, o tempo de computação. O segundo passo consiste em alinhar as restantes classes da O_s com as classes da segunda ontologia

alvo (O_{t2}), utilizando um método similar para a obtenção da semelhança. Por fim, criei também algoritmos de seleção que permitem escolher os mapeamentos com a maior semelhança associada, quando o alinhamento provisório tem mais do que um mapeamento para a mesma classe de O_s.

Para testar os algoritmos foram utilizadas nove ontologias diferentes, agrupadas em seis conjuntos de três ontologias. As O_s em todos os conjuntos possuíam definições lógicas no OBO Foundry. Estas definições lógicas foram criadas para ajudar à automatização do acesso a uma ontologia e para complementar definições textuais, que apenas podem ser interpretadas por utilizadores e não por computadores. Assim sendo, estas definições aprofundam a definição de cada classe e podem ser utilizadas como meio de melhorar a interoperabilidade entre diferentes ontologias.

As definições lógicas foram utilizadas como base para criar alinhamentos compostos de referência para ser possível avaliar automaticamente os algoritmos. Desta avaliação foi possível concluir que apesar dos algoritmos encontrarem um número significativo de mapeamentos corretos, os resultados ainda ficavam muito abaixo do esperado, uma vez que a precisão varia entre os 62.9 e os 11.7% e uma sensibilidade máxima de 60.7%. Por esta razão, fiz também uma avaliação manual que permitiu verificar que a maioria dos mapeamentos obtidos estavam corretos e a percentagem de mapeamentos incorretos era muito reduzida, com percentagens de corretos a variar entre os 44.9 e os 86.9% e os incorretos atingem o máximo de 17.6%, sendo que, num dos casos, a percentagem de incorretos foi 0. Estes resultados revelaram a incompletude dos alinhamentos de referência. No entanto, foi ainda encontrada uma percentagem considerável de mapeamentos que entravam em conflito (varia dos 9 até aos 50.4%) com os que estavam presentes no alinhamento de referência, ou seja, a mesma classe da ontologia de origem estava mapeada com classes alvo diferentes.

Esta análise manual permitiu verificar que a maioria dos mapeamentos encontrados se encontrava correto e permitiu também explorar as limitações do algoritmo, como a incapacidade atual para lidar com sinónimos que não se encontram explícitos na ontologia.

Devido ao elevado número de novos mapeamentos encontrados em relação aos alinhamentos de referência, inferi que uma das possíveis aplicações deste algoritmo poderia ser o auxílio na manutenção ou adição de novas definições lógicas e assim contribuir para uma melhor integração das ontologias presentes no OBO Foundry. Se esta aplicação fosse posta em prática poderia contribuir com mais de 900 novas definições lógicas.

As contribuições desta tese são originais dentro do domínio do alinhamento de ontologias e ainda existe muito espaço para desenvolvimento nesta área, uma vez que este é apenas um primeiro passo para a integração de dados biomédicos de forma mais complexa. Os algoritmos de alinhamento composto aqui propostos podem permitir não só acelerar o processo de manutenção de definições lógicas, como podem ser utilizados apenas pelas suas propriedades de alinhamento para fazer interrogações a bases de dados biomédicos e obter resultados mais complexos. Além disso, uma das contribuições deste trabalho foi a aplicação dos algoritmos para alinhar ontologias relacionadas com plantas, tarefa que teve um sucesso considerável, sendo que nenhuma modificação teve de ser implementada nos algoritmos para que este alinhamento funcionasse. Logo, esta aplicação permite extrapolar que é possível aplicar este algoritmo não só a ontologias biomédicas, como possivelmente a ontologias de outros domínios.

Por fim, o alinhamento de ontologias biomédicas poderá tornar-se ainda mais relevante nos próximos anos à medida que a necessidade de suportar análises cada vez mais complexas crescer, e se tornar necessário desenvolver os sistemas de forma a evoluírem e acompanharem este aumento de complexidade. Neste sentido, considero que esta dissertação é um primeiro passo para permitir o avanço da área e

para ajudar a expandir a utilização de ontologias biomédicas nas sua possíveis aplicações em bioinformática.

Acknowledgements

First, I would like to express my sincere gratitute to my advisor, Prof. Cátia Pesquita, whose expertise, understanding, and pacience, added considerably to my experience during this year. Yet, her guidance has been invaluable for some years now, seeing that she introduced me to the programming, which showed me, for the first time, how much I would like to do it in the future. Her availability and advise in my decisions and overwhelming support and encouragement throughout all aspects of my academic life (i.e., scholarship applications, cover letters, PhD applications) were crucial to my personal and academic development.

I'm grateful to Joana Barros and Catarina Martins for the many hours spent programming (i.e., solving bugs), discussing and writing (and despairing). Not only that but I am also thankful for the short walks and the long talks, which were the best way to relax, even during stressful times. Joana listened to all my nagging problems, but also provided me with the liveliest movie discussions ever. Catarina always brought the smile and the rabbit complaints. I am also grateful to Maria Fernandes for still caring about me and my work despite of being more than 2000km away.

I would also like to thank Fundação para a Ciência e Tecnologia, for awarding me a half year scholarship.

I am most of all grateful to my parents who supported my decision of studying Bioinformatics, despite still not quite getting what this is all about. Their understanding and continuous concern made me push myself to work harder everyday. Finally, to my sister, for the extremely important detours to relieve me early from work, for listening and for always letting the kid sister tag along.

Contents

List of Figures

List of Tables

List of Algorithms

List of Acronyms

AML AgreementMakerLight.

CO Crop Ontology.

EC Evidence Content.

FMA Foundation Model of Anatomy Ontology.

GO Gene Ontology.

ICBO International Conference on Biomedical Ontology.

NBO Neuro Behaviour Ontology.

OAEI Ontology Alignment Evaluation Initiative.

OBO The Open Biological and Biomedical Ontologies.

PATO Phenotypic Quality Ontology.

PO Plant Ontology.

SM String Matcher.

TO Plant Trait Ontology.

WBP WormBase Phenotype Ontology.

WM Word Matcher.

Chapter 1

Introduction

Research in the biomedical domain is generating massive amounts of data both due to high-throughput molecular biology techniques and studies, as well as the increasingly widespread adoption of health information systems that store clinical data. Despite this data deluge, the results of data analysis and the new knowledge derived from it is still mostly recorded in natural language, in the form of scientific publications, rendering its subsequent use a challenge. Moreover, the interoperability between biomedical databases is still a challenge as well, particularly when they belong to distinct domains.

A common strategy to deal with this significant knowledge and data expansion involves linking them to ontologies, making it easier to search through data in databases and to develop algorithms to process information. One of the most successful endeavours in this area was the application of the Gene Ontology (GO) (Ashburner *et al.*, 2000) to annotate huge amounts of new genomic data that is produced from high-throughput sequencing technologies. The success of the GO instigated the application of ontologies to a vaster array of biomedical knowledge to further enable information interchange and integration.

Currently there are multiple biomedical ontologies for specific disciplines in the life sciences. BioPortal, the largest biomedical ontologies repository, contains over 400 ontologies covering such distinct domains such as Zebrafish anatomy and Epidemiology. When databases employ the same ontology to annotate their data, they greatly simplify the process of interoperability. However, when data is annotated with distinct ontologies, interoperability is only possible if the ontologies

have links between them.

Indeed, there is already a substantial number of methods and tools for ontology matching, i.e., the establishment of relationships between semantically related entities (classes, properties or instances) from different ontologies. Ontology matching handles one of the paramount issues for Bioinformatics, the effective integration of data annotated with different ontologies. An alignment, which results from matching ontologies, can be used for a variety of tasks, including ontology merging, query answering, data translation or for browsing the semantic web (Euzenat *et al.*, 2007).

1.1 Motivation

Ontologies can be used to annotate different types of knowledge and are especially useful to query databases, since they help to standardize the knowledge and facilitate the recovery of annotated data. However, there are several biomedical ontologies for specific disciples in biomedicine, which were not constructed with a special focus on interoperability. Ontology matching can facilitate the process of finding relationships between equivalent terms of different ontologies, since most ontology matching systems produce equivalence mappings between classes or properties in two ontologies. Nonetheless, it may be advantageous to create mappings by combining entities from more than two ontologies to find even more complex relations.

For example, while in one ontology we can find the concept "Aortic Stenosis" (HP:0001650), in a second ontology we have the more general concept "Aorta" (FMA:3734) and yet in a third ontology we have the general descriptive term "Constricted" (PATO:0001847). "Aortic Stenosis" in the first ontology could then correspond to an "Aorta" in the second ontology which was "Constricted". Current ontology matching techniques would miss these cases, which would prevent ontology applications to use this information.

Therefore, to achieve a fuller integration of knowledge, we need to extend the current matching concept to cover more complex correspondences, which can support more sophisticated applications. Compound matching algorithms fulfil this need.

I argue that it would be useful to develop new techniques and tools for the identification of "compound matches", i.e. matches between class or property expressions involving more than two ontologies. To the best of my knowledge, there are currently no ontology matching systems capable of generating such mappings.

1.2 Objectives

The purpose of this work was to investigate whether existing ontology matching algorithms can be adapted or extended to support compound ontology matching, specifically ternary compound matching.

To achieve this goal I followed a strategy which divided the work in four main objectives:

1. Analyse a representative set of reference biomedical ontologies that would be used to test the future algorithms, explore the challenges I would be facing and to devise strategies to handle them.

2. Develop new algorithms for compound matching based on the adaptation or extension of existing classical matching algorithms, inspired by the previous step.

3. Evaluate the alignments obtained from the algorithms, both automatically, against reference alignments, in terms of their statistic reliability, and also manually, to better understand the results and point towards possible improvements.

4. Generalise the domain of application of the algorithms by proposing new candidate logical definitions and matching non-reference ontologies.

1.3 Contributions

The main contributions of this work are:

- **Novel compound matching algorithms.** Three new algorithms were created to produce compound alignments of significant quality. The first new algorithm follows a series of procedures to find equivalence between the three ontologies and creates the mappings which constitute the compound alignment. The second is a strict greedy selection algorithm, that given a compound alignment, saves to the final alignment the mapping with the higher similarity, when a source class appears more than once. The third is a permissive selection algorithm, which saves the top two mappings in the final alignment for source classes with more than one mapping. The results of these algorithms could allow the maintenance and discovery of several new candidate logical definitions for a set of ontologies. The algorithms were integrated with the AgreementMakerLight (AML) system and take advantage of its base lightweight structures to efficiently produce results. These algorithms are available on GitHub[1]. During the development of this work, the algorithms were used in an external project called The Planteome project[2] to align sets of three plant related ontologies.

- The preliminary results of the compound matching algorithms resulted in an extended abstract (Oliveira & Pesquita, 2015a), which was presented at the 2015 International Conference on Biomedical Ontology (ICBO), in Lisbon, in the Early Career track. Those results also encouraged the presentation of a poster (Oliveira & Pesquita, 2015b), which accessed the impact of the algorithms in the potential expansion of the logical definitions of biomedical ontologies.

- During this time I also participated in the Ontology Alignment Evaluation Initiative (OAEI), collaborating with the AgreementMakerLight team to improve this ontology matching system to participate in the 2015 competition.

[1] https://github.com/AgreementMakerLight/AML-Compound
[2] http://www.planteome.org/

- Finally, I collaborated in a related work with the DaSe Lab for Data Semantics at the Wright State University, USA (Cheatham & Hitzler, 2014), to test their new algorithm for property matching using AML's framework and to compare their results against AML's Property Matcher.

1.4 Overview

The following sections of the thesis describe all aspects of my work. Chapter 2 explains main concepts and tools applied in this research and also presents some related work. Chapter 3 presents an overview of the resources used, such as biomedical ontologies and the ontology matching system in which the algorithm was implemented. It also introduces the research questions and summarizes the methodologies employed. Chapter 4 and 5 describe methods, results and discussion of the two phases of development of the algorithm. Chapter 6 presents two practical applications of this work and, finally, chapter 7 summarizes the thesis and presents conclusions, debating some of the challenges found and how to address them in the future.

Chapter 2

Concepts and Related Work

This chapter is dedicated to describing some concepts necessary to contextualize this work, namely biomedical ontologies and ontology matching, as well as presenting some related work.

2.1 Biomedical Ontologies

Gruber (Gruber, 1993) defined an ontology as "an explicit specification of a conceptualization", i.e., ontologies provide conceptualizations of domains of knowledge and the specification is the concrete representation of that conceptualization. More formally, in information science, the word ontology is applied to a set of logic axioms that model a portion of reality (Guarino, 1998).

The main strength in applying ontologies to data from different fields is the ease with which researchers can share information and process data using computers. As such, ontologies describe knowledge in terms that can be understood by humans and machines alike.

An ontology can have different forms and different levels of complexity, but the backbone of an ontology is made of its concepts (often called classes). A concept can have a textual definition, a set of properties or even logical definitions. For example, the concept "person" can be defined by the sentence "an individual human being", which has the properties "name" and "date of birth" and can logically defined by the formula "Living entity ∩ Moving entity". A concept can also be defined by the set of instances which represent the application of the

concept with real world examples. For example, “Michael" is an instance of the concept “person”.

An ontology is organized with concepts from a given domain that have relationships between them. This organization forms a graph with each different node (concept) connected by a specific relationship, which describes interactions between concepts or properties of concepts.

Ontologies have been specially successful in the life sciences, since this field has a vast domain of knowledge and there is a need to develop tools to handle the large amounts of heterogeneous biological and clinical data. Therefore, biomedical ontologies such as the Gene Ontology (GO) (Ashburner *et al.*, 2000) and the Human Phenotype Ontology (HP) (Köhler *et al.*, 2013) were developed in order to address these problems. Figure 2.1 is a graph which represents part of HP where it is possible to see the different classes and their subclass relations. Biomedical ontologies have specific characteristics that need to be taken into account when developing tools and techniques to explore them:

- large size: biomedical ontologies commonly have thousands of classes, which can represent a computational challenge

- complex vocabulary: biomedical ontologies typically encode several names for the same class, including one main label and several synonyms of different kinds (e.g., narrow synonym, broad synonym).

- rich axioms: biomedical ontologies have been evolving towards greater semantic richness establishing different kinds of relations between classes (e.g., regulates, adjacent to, participate in) and complex axioms (e.g., ‘human patient’ and (has Age some float [>= 8]) *participant in* ‘WHO standard treatment for human brucellosis in adults and children eight years of age and older’).

Ontologies can help address a vast array of problems, for example, search and query of heterogeneous biomedical data, data exchange among applications, information integration, Natural Language processing, representation of encyclopedic knowledge and computer reasoning with data (Rubin *et al.*, 2008). Specifically,

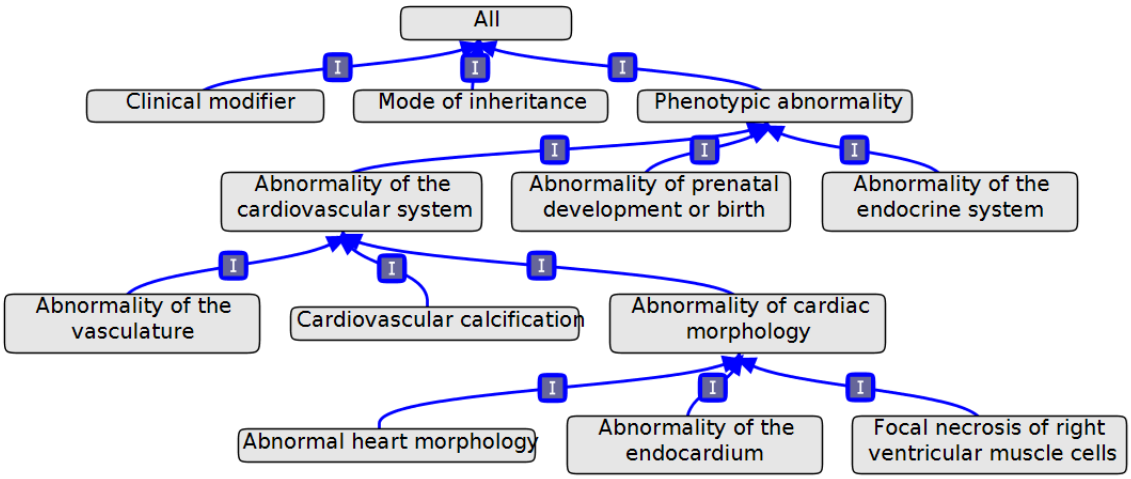

Figure 2.1: Graph representation of part of the Human Phenotype ontology, obtained using Obo-Edit (Day-Richter *et al.*, 2007).

biomedical ontologies can be tools for annotation and data integration that provide a common vocabulary to describe and communicate results. They also enable the creation of bioinformatics tools for analysis of microarray data, network modelling, and can also be applied to knowledge-based systems that include applications such as decision support in health care, which are typically dependent on large amounts of domain knowledge (Musen *et al.*, 2014).

2.1.1 Ontology languages: OBO and OWL

Ontologies provide a framework for computer reasoning, i.e., the specification of an ontology includes rules that can be automatically applied to items in a database in order to generate new knowledge. OBO and OWL are two ontology languages that provide a number of constructs for specifying such rules.

The Web Ontology Language (OWL) was developed by the W3C, the international standards organization for the Internet, as a language for defining structured, Web-based ontologies which enable richer integration and interoperability of data from multiple sources on the Web (Robinson & Bauer, 2011). An OWL ontology is a Resource Description Framework (RDF) graph, i.e., a set of RDF triples: a subject (identifies what the statement is about), predicate (the

property which the subject specifies) and object (value of that property) which together represent some piece of information.

The following is a representation of the beginning of a class of the Cell Type Ontology (CL) (Bard *et al.*, 2005) in OWL format. This partial representation shows the unique identifier of the class and its primary label. The full class representation can be found in Appendix A.

```
<owl:Class rdf:about="http://purl.obolibrary.org/obo/CL_0000007">
     <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">early
   embryonic cell</rdfs:label>
     ...
</owl:Class>
```

The OBO ontology language is presented in a relatively simple and human-readable format. The OBO flat file contains several stanzas which begin with the universal type `[Term]`, type definition `[Typedef]`, or instance `[Instances]`. The following is one example of a stanza in the CL OBO file:

```
[Term]
id: CL:0000007
name: early embryonic cell
namespace: cell
def: "A cell found in the embryo before the formation of all the
gem layers is complete." [GOC:tfm]
is_a: CL:0002321 ! embryonic cell
```

The stanza begins with the `id` which is an unique identifier to that term and it also has a name, i.e., a concise human-readable description of the term. Unique identifiers are commonly adopted by biomedical ontologies, encoded both in OBO or OWL. The `is_a` indicates semantic links of the ontology and, in the example above, we can see that the GO term `early embryonic cell` is a subclass of `embryonic cell`. The stanza can also include information such as synonyms and external references (`xref`). An important feature of OBO ontologies is the option to define a term using other terms from the same or different ontologies.

These logical definitions are constructed using the keyword `intersection_of` and can be used for computer-based reasoning.

OBO ontologies have been developed by or in close collaboration with biomedical domain experts and are designed to be readable by human and machines. OWL ontologies are often developed by computer scientists, which focus on developing frameworks, algorithms and software for inference (Robinson & Bauer, 2011) and are not designed with human readability in mind.

2.1.2 OBO Foundry

In order to respond to the increasingly heterogeneous ontologies, The Open Biological and Biomedical Ontologies (OBO) Foundry was created (Smith *et al.*, 2007) to provide some coordination in the establishment of interoperability. This open repository stores a variety of ontologies developed for sharing information across different biological and medical domains and currently has more than 100 ontologies. Their mission is defined as follows[1]:

> The mission of OBO is to support community members who are developing and publishing ontologies in the biomedical domain. It is our vision that a core of these ontologies will be fully inter-operable, by virtue of a common design philosophy and implementation, thereby enabling scientists and their instruments to communicate with minimum ambiguity. In this way the data generated in the course of biomedical research will form a single, consistent, cumulatively expanding, and algorithmically tractable whole. This core will be known as the "OBO Foundry".

Despite this vision, there are currently only nine ontologies that follow OBO guidelines in full, which means that for the remaining hundreds of ontologies, achieving consistency and interoperability is still a challenge.

[1]Retrieved from http://www.obofoundry.org/ on 9 June 2015

2.1.3 Logical Definitions

One notable effort in increasing the interoperability of ontologies has been the creation of logical definitions. Almost all classes in a biomedical ontology have a textual definition, which can be interpreted by a human user, but can't be easily accessed by a computer without sophisticated natural language processing. Therefore, efforts have been made to transform these definitions into a computable form as a set of logical definitions.

Therefore, logical definitions aid with the automatization of the access to an ontology and complement text definitions. They could also be potentially used to reason over an ontology or to automatically derive relationships between classes.

Logical definitions are applied to classes and use genus-differentia constructs of the form "X is a G that D", where X is the defined class, G is the genus and D the differentia. The genus is a more general class than X and the differentia discriminates instances of X from other instances of G (Mungall *et al.*, 2011). Figure 2.2 shows an example of a logical definition.

```
[Term]
id: MP:0000216 ! absent erythroid progenitor cell
intersection_of: PATO:0002000 ! lacks all parts of type
intersection_of: inheres_in CL:0000038 ! erythroid progenitor cell
```

Figure 2.2: Example of a logical definition of a GO term defined in OBO.

However, creating, implementing and maintaining these computable definitions can be difficult, as it requires a lot of manual labour and it is unclear if the cost is worth the trouble (Mungall, 2004).

To help address this challenge, strategies to automate part of the process can be applied and Obol (Mungall, 2004) was designed with this purpose in mind. Obol uses a set of fairly complex ontology-specific grammar rules to generate proposed logical definitions from pre-existing classes, which are then vetted by experts. So, Obol is manly used as an aid to the addition and curation process. It has been applied in the improvement of phenotype ontologies (Mungall *et al.*, 2010) and in the normalization of GO (Mungall *et al.*, 2011).

2.2 Ontology Matching

Ontology matching is a solution to the problem of heterogeneity between different ontologies. It enables computer systems to establish connections between ontologies with similar domains. Euzenat *et al.* (2007) defines the matching process as a function f which, from a pair of ontologies o and o', returns an alignment A' between these ontologies. Therefore, an alignment consists of a set of correspondences (mappings) between semantically related entities of different ontologies. This process can be extended by using an input alignment A and by using other parameters p, e.g., weights, thresholds, and even external knowledge r.

$$A' = f(o, o', A, p, r) \tag{2.1}$$

It can be schematically illustrated by Figure 2.3.

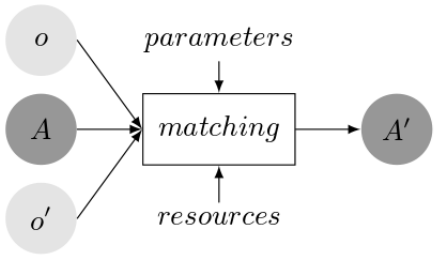

Figure 2.3: The matching process.

The algorithms which perform the match are commonly called matchers and they usually assign a confidence level to each mapping, which reflects the similarity between classes. Correspondences between ontology entities are usually of equivalence, but can also include consequence, subsumption, or disjointness.

2.2.1 Compound Ontology Matching

Currently, an ontology alignment is predominantly obtained as mappings between two ontologies, i.e., binary matching. However, the concept of Compound Matching was recently defined in Pesquita *et al.* (2014), which focused specifically on ternary compound matching.

A ternary compound alignment is a set of mappings between classes from a source ontology O_s and class expressions obtained by combining two other classes, each belonging to a different target ontology O_{t1} and O_{t2} (See Figure 2.4). This means that a ternary compound mapping is a tuple <X, Y, Z, R, M>, where X, Y and Z are classes from three distinct ontologies, R is a relation established between Y and Z to generate a class expression that is mapped to X via a mapping relation M. The ontology to which X belongs is considered to be the source ontology, and the ontologies that define Y and Z are considered as the target ontologies 1 and 2, respectively. In this particular case, the relation R is always an intersection (regardless of any qualifier) and the mapping M an equivalence.

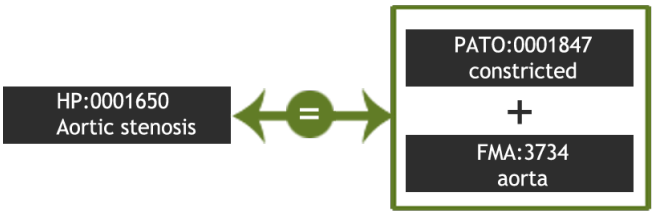

Figure 2.4: Example of a possible ternary compound mapping between the Human Phenotype, Foundational Model of Anatomy and Phenotypic Quality ontologies.

2.3 Ontology Matching Systems

Euzenat *et al.* (2007) defined two classifications to distinguish matching techniques based on the most relevant proprieties in the ontology matching domain. They defined the *Granularity/Input Interpretation*, which distinguishes between element-level and structure-level matching techniques.

The element-level matching techniques ignore relations between entities or their instances and focus on analysing them in isolation. Structure-level techniques find matches by analysing not only entities or their instances, but the structure as a whole.

2.3.1 Element-level techniques

Euzenat *et al.* (2007) defines six different element-level matching techniques:

- **String-based techniques:** strings are considered as a sequence of letters in an alphabet and usually the more similar the strings, the more likely they are to represent the same concept. There are several ways to compare the structure of two strings, such as (1) normalisation to reduce strings to a common format, (2) using substrings to base the similarity between strings in common letters, (3) edit distances to evaluate how a string can be a variation of another, or (4) statistical measures that weigh the importance of one word in relation to others.

- **Language-based techniques:** names are considered as words in some natural language and use natural language processing techniques to exploit morphological properties of the input words.

- **Constraint-based techniques:** the algorithms deal with entities which have internal constraints, such as types, cardinality of attributes or keys.

- **Linguist resources:** the matching techniques use linguist resources such as lexicons or domain specific thesauri to match words based on linguist relations such as synonyms or hyponyms.

- **Alignment reuse:** by taking advantage of previously matched ontology alignments, it is possible to have different combinations of mappings in future alignments. If the ontologies are describing the same application domain, previous matches can be similar to future matches that use the same ontologies.

- **Upper level and domain specific formal ontologies:** upper level ontologies, which are logic-based systems, can be used as external sources and the matching systems that could use them are semantic based. Domain specific formal ontologies can also be used as background knowledge, since they focus only on one particular domain and can be used to help match poorly structured ontologies.

2.3.2 Structure-level techniques

- **Graph-based techniques:** the ontologies are considered as graphs and the similarity is obtained by the analysis of the positions of the nodes within the graphs.

- **Taxonomy-based techniques:** these are also graph based, but also consider the specialisation relation, since a is_a relation links terms already similar.

- **Repository of structures:** similarities between structures are stored and when a new match is being made they are first checked for similarity against the structures in the repository.

- **Model-based techniques:** the algorithms handle the input based on its semantic interpretation, i.e., if two entities are the same, then they share the same interpretations.

- **Data analysis and statistics techniques:** these techniques take advantage of a representative sample of a population in order to find regularities and discrepancies to group items or to compute distances between them.

2.3.3 Ontology Matching Tools

So far there have been developed at least 60 different ontology matching systems (Otero-Cerdeira *et al.*, 2015) and while some have been discontinued, there are several still receiving constant updates and enhancements. In the following section I will briefly describe four popular ontology matching systems that perform well on biomedical ontologies.

2.3.3.1 AgreementMaker

AgreementMaker (Cruz *et al.*, 2009) is a schema and ontology matching system. It allows a wide range of input and output formats and includes several matching methods depending on several parameters which rely on user input. This system

was first created in 2007 and has since received several updates and enhancements. Overall, AgreementMaker has a flexible and extensible framework with a comprehensive user interface.

2.3.3.2 AgreementMakerLight

Since AgreementMaker was not designed to match ontologies with more than a few thousands of concepts, AgreementMakerLight (AML) (Faria *et al.*, 2013) was developed as a novel ontology matching framework, derived from AgreementMaker, with a focus on the efficient matching of very large ontologies. AML is not designed to support user interaction, but maintains the flexibility and extensibility of the AgreementMaker. AML was created in 2013 and it has been in continuous development ever since, being currently the best performing system for the alignment of biomedical ontologies.

2.3.3.3 LogMap

LogMap (Jiménez-Ruiz & Cuenca Grau, 2011) is based on a set of anchor mappings obtained from lexical comparison and it alternates between a repair and a discovery step, in order to find new mappings. To discover the new anchors structural information is also exploited. LogMap was created in 2011 and has achieved a high level of maturity in these last years, with continuous publications describing its performance and improvements.

2.3.3.4 YAM++

YAM++ (Ngo & Bellahsene, 2012) uses machine learning techniques to align ontologies. It uses two matchers at element and structural level to discover new mappings. The mappings resulting from the combination of these two matchers are then revised by a semantical matcher in order to remove inconsistent mappings. The system was first presented in 2009 and has received a few updates since.

2.3.4 Ontology Alignment Evaluation Initiative

The Ontology Alignment Evaluation Initiative (OAEI)[1] is a coordinated international initiative which showcases, compares and evaluates an increasing number of ontology matching systems to draw conclusions about the best matching strategies. It was created in 2004 and since then takes place, annually, around the world. Their ambition is that, from these evaluations, developers can improve their systems.

Currently, the initiative contains eight different tracks:

- **Benchmark:** the goal is to align data sets that are built from reference ontologies to identify the strengths and weaknesses of matchers and test if they are reusable over the years.
- **Anatomy:** uses real world cases by align the Adult Mouse Anatomy (2744 classes) and the NCI Thesaurus (3304 classes) describing the human anatomy.
- **Conference:** using a collection of ontologies describing the domain of organising conferences, the goal is to evaluate the alignments against one or more reference alignments.
- **Multifarm:** uses a subset of the Conference dataset, translated in nine languages to compare the performance of matching approaches with a special focus on multilingualism.
- **Large Biomedical Ontologies:** this track evaluates the alignment of the Foundation Model of Anatomy Ontology (FMA), SNOMED CT and National Cancer Institute Thesaurus (NCI) ontologies. These ontologies are semantically rich and contain tens of thousands of classes.
- **Interactive matching evaluation:** the goal of this track is to show if user interaction can improve matching results, using the Conference, Anatomy and Large Biomedical ontologies datasets.

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

- **Instance Matching:** evaluates the performance of detection of the degree of similarity between pairs of items/instances
- **Ontology Alignment for Query Answering:** evaluates alignments by their ability to enable query answering in an ontology based data access scenario, where multiple aligned ontologies exist.

All of the previously referenced ontology matching systems have participated in the OAEI before and achieved good results in the years they participated. The 2014 edition had 14 participants which account for 14 different strategies for ontology matching (Dragisic *et al.*, 2014). AML ranked first first in 6 of the 8 tracks and was its best performance yet.

2.4 Related Work

The first reference to compound ontology matching was made in 2014 by Pesquita *et al.* (2014). Their aim was to stimulate the development of new techniques which increased the complexity of the mappings, by aligning more than two ontologies. Since this is a new area in the ontology matching field they proposed a way to create a benchmark to measure a system's performance, by using OBO cross-products to create compound reference alignments.

They also tested some preliminary strategies for compound matching. Their strategy involved first matching the source ontology to each of the target ontologies individually, using an [U+FFFD]anchor[U+FFFD]-based word matching algorithm, and then matching all pairs of target classes that map individually to the same source class. Their implementation was integrated into AML, and made use of its WordMatcher. To measure similarity between the source labels and merged target labels they employed a modified Jaccard index as a similarity metric. However, despite the reduction of the search space by employing this strategy, they still could not test their algorithm with larger sets of ontologies. They tested the strategy in the MP-PATO-CL and MP-PATO-NBO alignments, obtaining recall values of 30 and 11% respectively, but precision values below 1%.

Despite this being the first reference to compound matching, the closely related concept of complex ontology matching was first addressed in 2003 (Xu &

Embley, 2003) and it can be considered similar in some ways to compound ontology matching.

Complex ontology matching is commonly defined as a correspondence between two classes from different ontologies, where one of them is a complex concept or property description. It differs from compound matching because it still only uses two ontologies, but it has in common the purpose of aligning more than two entities in those ontologies. A complex mapping can be defined as (Hu *et al.*, 2012):

$$X : \mathcal{O}_1 \leftarrow Y_1 : \mathcal{O}_2 \wedge Y_2 : \mathcal{O}_2 \wedge \ldots \wedge Y_n : \mathcal{O}_2 \tag{2.2}$$

where X, Y_j (j = 1, 2, . . . , n) are classes or properties in $\mathcal{O}_1$ and $\mathcal{O}_2$, respectively.

Whereas a ternary compound mapping with the same notation would be:

$$X : \mathcal{O}_1 \leftarrow Y_1 : \mathcal{O}_2 \cap Z_1 : \mathcal{O}_3 \tag{2.3}$$

where X, Y and Z are classes or properties in $\mathcal{O}_1$, $\mathcal{O}_2$ and $\mathcal{O}_3$, respectively.

As exemplified in Ritze *et al.* (2009): "While in one ontology we have an atomic concept *AcceptedPaper*, in another ontology we have the general concept *Paper* and the boolean property *accepted*. An *AcceptedPaper* in the first ontology corresponds, in the second ontology, to a *Paper* that has been accepted."

To solve the complex ontology matching problem, Ritze *et al.* (2010) used a pattern-based approach, where they present correspondence patterns and define matching conditions for each of them. They use linguistic analysis to base these conditions and achieve a significantly higher precision than the one obtained by their previous approach (Ritze *et al.*, 2009), which used a simple string-based similarity value.

Doan *et al.* (2003) developed a system, called GLUE, which applies machine learning techniques to semi-automatically create semantic mappings. This system is used for 1:1 ontology alignments, however, they also created the CGLUE which is used to find complex mappings. Here they define a complex mapping as a match between two nodes of a taxonomy, where the node of the target ontology usually is a set of instances connected by an operator, such as union. They use a similar approach for both the simple and complex mappings, however, due to the

exponential number of nodes, they adapt the beam search technique, commonly used in Artificial Intelligence, to search only through a pre-determined number of the most promising candidates.

Finally, Dhamankar *et al.* (2004) defines complex ontology matching as a set of attributes in one schema which correspond to a combination in another. They developed a system, called iMAP, to semi-automatically discover both one-to-one and complex matches. They used two techniques, first they use several searchers, each considering a meaningful subset of the space. As examples they present a text searcher that may consider only matches that are concatenations of text attributes, while a *numeric searcher* considers arithmetic expressions. Then they use beam search to control the search through the space of candidate matches. Still the number of match candidates is often infinite and so they developed a termination criterion based on the diminishing-returns principle to stop the search. To evaluate the quality of each match candidate, they employed a set of techniques, including machine learning, statistics, and heuristic methods.

Some of the strategies employed by these works, such as filtering, patterns and dealing with large search spaces helped guide the ideas and strategies which were employed during the development of this work.

Chapter 3

Resources and Methodology

In the following section, I describe the main resources used in this work and present a schematic view of the research questions and methodologies applied to answer them.

3.1 Resources

I will now detail three different kinds of resources previously and independently developed which were used in the course of my work, which are:

- A set of reference Biomedical ontologies
- An ontology matching system, AgreementMakerLight
- A set of compound reference alignments

3.1.1 Biomedical Ontologies

The following ontologies are a part of the OBO Foundry and were used throughout this work. They were chosen due to their presence in the logical definitions of MP, HP and WBP and, therefore, in the reference alignments.

These ontologies were downloaded from the OBO Foundry in May 2015 (http://obo.sourceforge.net).

Table 3.1: Biomedical ontologies.

Ontology	Acronym	Classes	Names	Reference
Cell Type	CL	4775	4375	Bard *et al.* (2005)
Foundational Model of Anatomy	FMA	78977	126190	Rosse & Mejino (2003)
Gene Ontology biological process domain	GO	43048	276577	Ashburner *et al.* (2000)
Human Phenotype	HP	28621	18431	Köhler *et al.* (2013)
Mammalian Phenotype	MP	28643	29592	Smith *et al.* (2004)
Neuro Behaviour Ontology	NBO	116710	1168	Gkoutos *et al.* (2012)
Phenotypic quality	PATO	2497	3378	Mungall *et al.* (2010)
Uber Anatomy Ontology	UBERON	18322	50713	Haendel *et al.* (2009)
***Caenorhabditis elegans* phenotype**	WBP	2290	2739	Schindelman *et al.* (2011)

Table 3.1 presents the number of different classes and names that each of these biomedical ontologies contain. The 'Names' column represents the number of labels and synonyms that each ontology possesses. If the number of names is lower than the number of classes, the ontology has imported classes from other ontologies. The GO ontology has the higher number of names and so, if a lexical alignment is performed and all synonyms are considered, any alignment using

this ontology will be inherently harder to compute. The NBO ontology has the least amount of names, despite having many classes. This means that it contains several classes with no direct label which originate from imported classes. It was not this work's goal to handle imported classes, so these classes were not considered in the development of the compound matching algorithms.

Therefore, from the average number of names of these ontologies it is possible to conclude that this work involves handling large sets of ontologies, which could be an issue if not addressed correctly during the development process.

3.1.2 Reference Alignments

The construction of the compound reference alignments originated from a previous work (Pesquita *et al.*, 2014) where logical definitions[1] of OBO ontologies were used to derive ternary compound mappings to be used as a gold-standard. They selected OBO ontologies with over 100 logical definitions that could be converted to ternary compound mappings. This means that the logical definitions need to provide an equivalent class expression by intersecting two classes from two other ontologies. Figure 3.1 details a ternary compound mapping encoded in an extension of the Alignment API RDF format for the logical definition presented in Figure 2.2.

```
<align:map>
    <align:Cell>
        <align:entity1>
            <edoal:Class rdf:about="http://purl.obolibrary.org/obo/MP_0000216"/>
        </align:entity1>
        <align:entity2>
            <edoal:Class>
                <edoal:and rdf:parseType="Collection">
                    <edoal:Class rdf:about="http://purl.obolibrary.org/obo/CL_0000038"/>
                    <edoal:Class rdf:about="http://purl.obolibrary.org/obo/PATO_0002000"/>
                </edoal:and>
            </edoal:Class>
        </align:entity2>
        <measure rdf:datatype="xsd:float">1.0</measure>
        <relation>=</relation>
    </align:Cell>
</align:map>
```

Figure 3.1: Ternary compound mapping encoded in an extension of the alignment API RDF format. `entity1` represents the source class, `entity2` contains the target 1 and target 2 by order of appearance.

[1] Retrieved from http://www.obofoundry.org/index.cgi?show=mappings

Following the previous rules, seven sets of possible gold-standard compound alignments were defined. In this work, six of the seven references were used. It was necessary to exclude the one alignment due to its disparities in defining logical definitions when compared with the other six reference alignments.

Therefore, for all of the subsequent tests the sets of ontologies shown in Table 3.2 were used.

Table 3.2: Sets of ontologies and respective number of unique classes present in the reference alignments.

Source	No. classes	Target 1	No. classes	Target 2	No. classes
MP	474	CL	198	PATO	32
MP	944	GO	576	PATO	58
MP	219	NBO	146	PATO	17
MP	1999	UBERON	785	PATO	203
WBP	324	GO	236	PATO	40
HP	1894	FMA	640	PATO	187

Table 3.2 presents an analysis of the reference alignments. Each of the ontologies has an associated number of classes which indicates how many different classes of that ontology are present in that particular alignment. The number of classes of the source ontology also represents the total number of mappings present in the reference, since they were created from logical definitions, where each class is only represented once, making this an alignment with a final cardinality of 1.

This table also shows that several classes are used repeatedly in the compound mappings, specially classes from the PATO ontology. So, it is possible to conclude that the mappings present in the reference alignments are combinations of a restrict number of target classes, since there is considerably less diversity in the target ontologies than the source ontology.

3.1.3 AgreementMakerLight

The algorithms developed in this work were integrated into the AgreementMakerLight (AML) (Faria *et al.*, 2014a) ontology matching system. This system was

developed to handle large ontologies and is currently the most successful system in the alignment of biomedical ontologies, which makes it a suitable framework for the proposed compound matching algorithms.

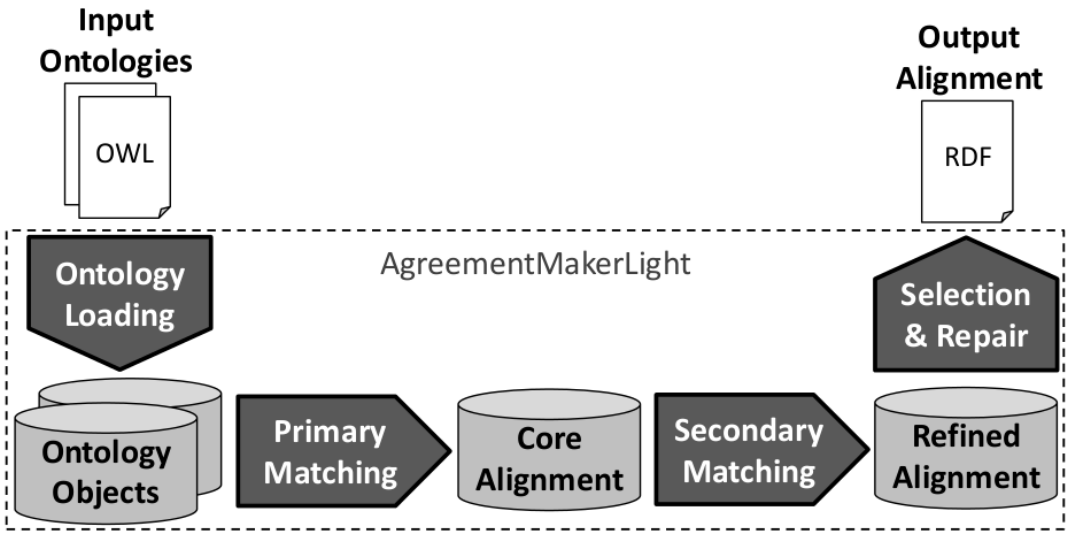

Figure 3.2: AgreementMakerLight ontology matching framework.

AML's core framework, represented in Figure 3.2, is composed of three modules (Faria *et al.*, 2013): the ontology loading module, the ontology matching module and alignment selection and repair. The ontology loading module reads the input ontologies and builds core objects like the *Lexicon*, which is a table of class names and synonyms in an ontology, or the optional *RelationshipMap* which saves `is_a` and `part_of` relationships and all disjoint clauses.

AML's matching module features several matchers to tackle different ontology matching problems. AML divides its matchers into *Primary Matchers* and *Secondary Matchers*. Primary Matchers are matching algorithms that rely on HashMap cross-searches, which take $O(n)$ time. Secondary Matchers make non-literal comparisons between terms and require an all-against-all comparison of the ontologies, which takes $O(n^2)$ time. Secondary Matchers, therefore, are not the ideal choice for large ontologies.

AML is more focused on lexical matching techniques with matchers such as:

- the *Lexical Matcher*, a simple and efficient algorithm which makes literal name matches between ontologies.
- the *String Matcher (SM)*, which was directly ported from AgreementMaker and includes several similarity metrics. It makes non-literal comparisons and therefore is a secondary matcher.
- the *Word Matcher (WM)*, a word-based string similarity algorithm that compares each word of the label of a class and ponders their similarity through a weighted Jaccard index. It is based on the Multi-word Matcher of AgreementMaker, but instead of an all-against-all comparison, it is based on a lexical cross-search. This matcher creates a derivation of the *Lexicon* and computes the frequency and Evidence Content (EC) for each word, instead of the whole label. The EC of each word is given by the inverse logarithm of its frequency in the lexicon. Despite of being a Primary Matcher, the Word Matcher is more memory intensive than other lexical matchers, since it stores a temporary alignment between all classes that share at least one word and, for that reason, ontologies are partitioned during the matching process. The algorithm uses a bag-of-words approach[1] to compute the similarity between two classes.

The alignment selection and repair module ensures that the final alignment has the desired cardinality and that there are no violations of restrictions of the ontologies.

3.2 Methodology

The scheme presented in figure 3.3 represents the overall flow of this work, with research questions and a summary of the employed methodologies for each for the three tasks.

[1]The label is represented as a multiset of its words, disregarding word order and grammar.

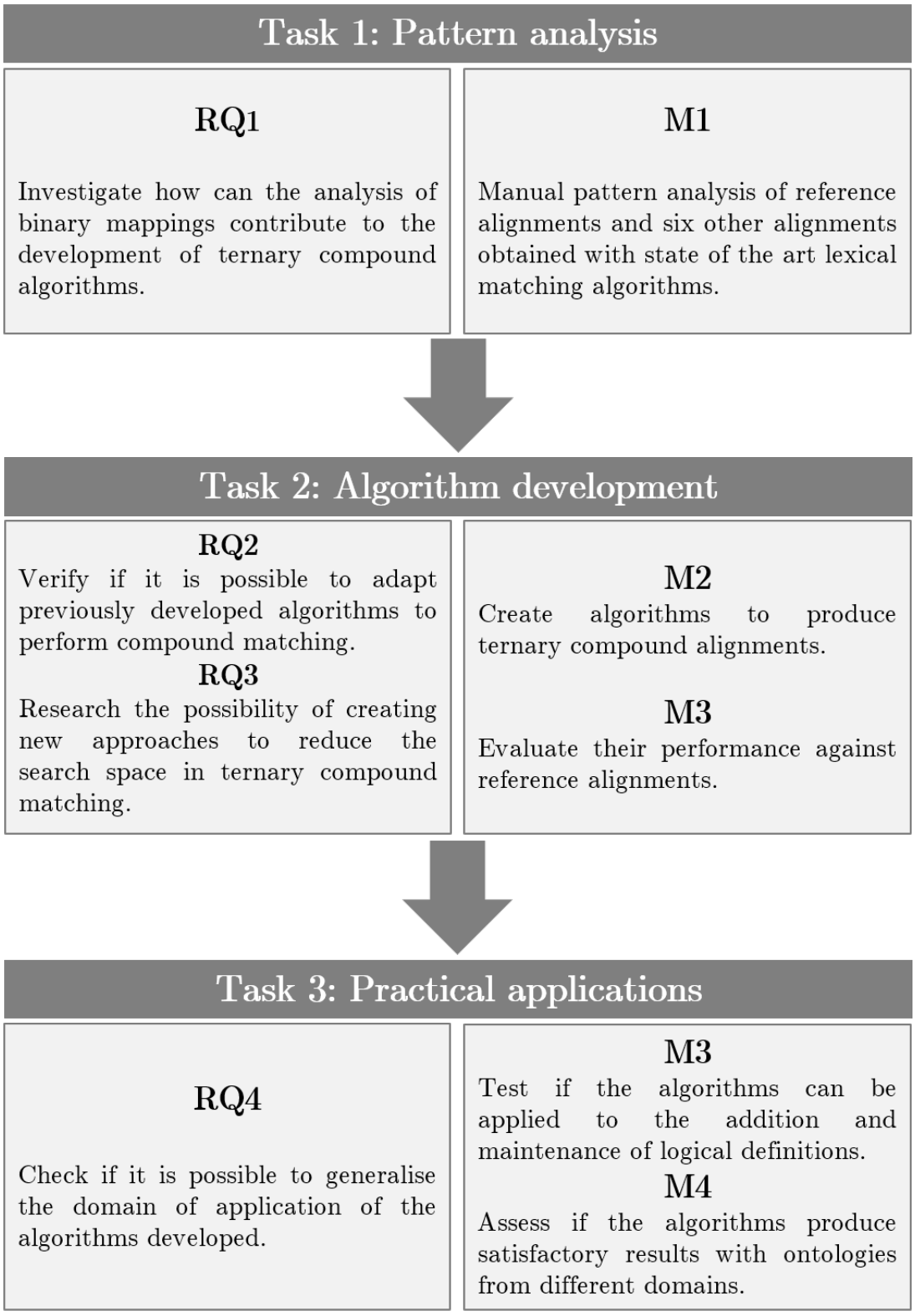

Figure 3.3: Methodology scheme. RQ: Research Question. M: Methodology

Chapter 4

Pattern Analysis of Lexical Matching

This chapter is dedicated to the first task of this work, where an exploratory analysis of lexical patterns in binary mappings served as a conceptual foundation for the development of compound matching algorithms in task 2.

4.1 Introduction

In ontology matching, a common approach to find relevant mappings is to compute similarities between all classes of each of the ontologies. In standard binary matching problems, this leads to a quadratic search space, and so state-of-the-art ontology matching systems, such as AML and LogMap, employ “anchor"-based strategies to increase efficiency. In ternary ontology matching, the search space is cubic, so matching even relatively small ontologies can pose efficiency problems, since we would need to compute all combinations of labels from both target ontologies.

The work by Pesquita *et al.* (2014) tried to reduce the search space by first matching the source ontology to each to the target ontologies individually, using an “anchor"-based word matching algorithm, and then matching only all pairs of target classes that map individually to the same source class. However, in spite of the reduced search space, they were not able to test this matching strategy on

sets that included larger ontologies and for the ones which produced compound alignments, they obtained low recall values and precisions below 1%.

Therefore, the first task of this work had an exploratory nature and its goal was to better understand the mappings between source and target ontologies and to seek new strategies to apply to ternary compound matching. This task consisted in analysing two sets of alignments: the reference alignments and binary alignments produced by AML from the 6 sets of biomedical ontologies previously defined (see Table 3.2), using the source ontology and a single target ontology. The tests were performed on these two collections of alignments because the reference alignments have a low coverage of source classes, and may not showcase all kinds of patterns.

4.2 Lexical Patterns

4.2.1 Methods

For each of the 6 sets of ontologies two alignments were produced, one using AML's Word Matcher (WM) and the other using String Matcher (SM), with an empirically defined 0.6 threshold.

Then the obtained alignments were manually analysed to uncover lexical patterns. The patterns found were categorized into different types. I also applied the same type of analysis to the compound reference alignments, once again considering only source-target mappings.

4.2.2 Results

I obtained two alignments for each pair of ontologies, one from the String Matcher and the other from the Word Matcher. The total number of mappings obtained in each of the alignments is presented in table 4.1

Table 4.1: Number of Mappings for each pair of ontologies.

Source	Target	SM Mappings	WM Mappings
MP	CL	34	5
MP	GO	501	65
MP	NBO	594	219
MP	UBERON	71	50
WBP	GO	322	219
HP	FMA	304	252
MP	PATO	29	59
WBP	PATO	41	25
HP	PATO	25	12

4.2.2.1 Pattern Types

The following four patterns emerged from the analysis:

1. **Addition**: the source or target class label has one or more extra words. The word can be at the beginning, the middle or the end of the label. Table 4.2 shows three examples of this type of pattern.

Table 4.2: Addition pattern examples.

Source URI	Source Label	Target URI	Target Label
MP:0003214	neurofibrillary tangles	GO:1902988	neurofibrillary tangle assembly
HP:0005819	short middle phalanx of finger	FMA:35483	phalanx of middle finger
WBP:0001911	axon regeneration defective	GO:0031103	axon regeneration

2. **Variation**: The number of words is the same, but one word does not match.

Typically it is a modified suffix or prefix, different number or a relevant combination of numbers and letters with no more than two characters. Table 4.3 shows examples of this type of pattern.

Table 4.3: Variation pattern examples.

Source URI	Source Label	Target URI	Target Label
MP:0012818	rhombomere transformation	GO:0021594	rhombomere formation
MP:0002269	muscular atrophy	GO:0014889	muscle atrophy
HP:0011903	hemoglobin h	FMA:72160	hemoglobin a2

3. **Combination:** table 4.4 shows three examples of mappings which contain a combination of the previous patterns.

Table 4.4: Combination of patterns examples.

Source URI	Source Label	Target URI	Target Label
MP:0009056	abnormal interleukin 21 secretion	GO:0072634	interleukin 30 secretion
HP:0001065	striae distensae atrophy	FMA:76746	stria
MP:0013527	absent conjunctiva goblet cells	CL:2000084	conjunctiva goblet cell

4. **Full match**: table 4.5 shows mappings which have labels that match completely, but can sometimes have words in a different order.

Table 4.5: Full match pattern.

Source URI	Source Label	Target URI	Target Label
MP:0009461	skeletal muscle hypertrophy	GO:0014734	skeletal muscle hypertrophy
WBP:0001392	cell cycle arrest	GO:0007050	cell cycle arrest
MP:0002119	dipsosis	NBO:0000541	dipsosis

There were some mappings which had no discerning pattern and could not fit in any of the patterns mentioned above. Examples of this kind of mappings are presented in table 4.6.

Table 4.6: Mappings with no discerning pattern.

Source URI	Source Label	Target URI	Target Label
MP:0000250	abnormal vasoconstriction	GO:0003056	regulation of vascular smooth muscle contraction
MP:0002555	addiction	PATO:0002133	adduction
MP:0002229	neurodegeneration	GO:0070657	neuromast regeneration

The mappings in Table 4.1 are a representation of the mappings which could not fit in any other pattern. They are composed mostly of incorrectly matched mappings or mappings which involve synonyms.

4.2.2.2 Reference Alignments

The second part of this task involved performing a manual analysis of the reference alignments. I concluded that the majority of the mappings fit in at least one of the categories previously defined and most are a combination of the first two patterns. Table 4.7 shows examples of these mappings and indicates to which pattern the mapping belongs.

Table 4.7: Reference mappings with the corresponding pattern.

Source URI	Source Label	Target URI	Target Label	Type
HP:0000003	Multicystic kidney	FMA:7203	Kidney	1
MP:0001237	enlarged spinous cells	CL:0000649	prickle cell	3
MP:0000262	poor arterial differentiation	GO:0048844	artery morphogenesis	3

However, Table 4.8 presents a new pattern which is the occurrence of synonyms between the two classes that are being matched.

Table 4.8: Synonym based pattern

Source URI	Source Label	Target URI	Target Label
MP:0000422	delayed hair appearance	GO:0042640	anagen
MP:0000127	degenerate molars	UBERON:0003655	molar tooth
HP:0010108	Aplasia of the hallux	FMA:25047	Big toe

4.2.2.3 Pattern Count

I developed a simple script to count the number of mappings which would fit in the first and/or second pattern types and table 4.9 shows these results.

Table 4.9: Distributions of mappings fitting lexical patterns 1 or 2.

Matcher	Ontology	Addition Pattern	Variation Pattern	Size
String Matcher	MP-CL	26	7	34
	MP-GO	287	210	501
	MP-NBO	354	205	594
	MP-UBERON	58	11	71
	WBP-GO	182	137	322
	HP-FMA	272	23	304
	MP-PATO	18	1	29
	WBP-PATO	28	2	41
	HP-PATO	12	1	25
Word Matcher	MP-CL	4	1	5
	MP-GO	32	25	65
	MP-NBO	118	44	219
	MP-UBERON	42	5	50
	WBP-GO	183	33	219
	HP-FMA	158	44	252
	MP-PATO	33	21	59
	WBP-PATO	19	1	25
	HP-PATO	6	0	12
Reference	MP-CL	439	12	474
	MP-GO	805	83	944
	MP-NBO	177	24	219
	MP-UBERON	1693	126	1999
	WBP-GO	256	39	325
	HP-FMA	1691	66	1893
	MP-PATO	3096	35	3636
	HP-PATO	1710	8	1893
	WBP-PATO	302	4	325
	Total	12001	1168	

It is possible to see from table 4.9 that most of the mappings can fit in the addition pattern (12001), while 1168 mappings belong to the variation pattern.

4.3 Discussion

Through the manual analysis of the automated alignments, four different types of patterns were defined, which could be found in most of the reference alignments. The reference alignments also contained an additional pattern, the synonym-based. This last pattern was not found by AML's matchers mainly because some synonyms are not explicitly expressed in the ontology.

This exploratory task led to four findings that may have an impact on designing compound matching algorithms for biomedical ontologies:

- The 'addition' pattern reveals that a majority of mappings include adding or removing words from the classes labels, indicating that a bag-of-words approach is likely an efficient solution
- The 'variation' pattern can be addressed by the use of a stemmer (see chapter 5, section 5.1.2.1 for more details). For example, for the 'muscular atrophy' (MP:0002269) and 'muscle atrophy' (GO:0014889) classes referenced in Table 4.3, a stemmer could probably reduce the first word to 'musc' and both terms would be labelled 'musc atrophy' and they would match.
- Some mappings, such as the second example from table 4.2, can belong to a specific pattern, but have their word order changed. This order variation could modify the meaning of the class, and lead to some errors when applying a bag-of-words approach.
- The 'synonym' pattern could prove to be a challenge, because a direct bag-of-words word matching algorithm will not find those types of matches. Employing external resources such as WordNet should be considered.

This evaluation also led to the discovery of some issues with the base framework, since AML was ignoring the words in the class label that had numbers and/or words with two or less letters and, thus, was not correctly matching any of those.

Chapter 5

Compound Ontology Matching Algorithms

The findings obtained by the exploratory analysis of the alignments detailed in the previous chapter served as the conceptual foundation for the development of the compound matching algorithms described in this chapter.

The prevalence of the 'addition' pattern led to the idea that the most effective method would be to employ a sequential approach, where in a first step the source and one of the target ontologies are matched using a bag-of-words strategy. For example, the class 'aortic stenosis' (HP:0001650) would be matched to 'aorta' (FMA:3734). In the second step, the remaining source word 'stenosis' would be matched to the other target ontology.

The remainder of this chapter details the algorithms developed for compound matching and selection, and their extensions using stemming, WordNet and a string matching algorithm.

Results for each experimental investigation and evaluation method are presented and discussed independently. An overall discussion is presented at the end of the chapter.

5.1 Methods

5.1.1 Compound Matching Algorithm

I developed a novel algorithm to establish compound mappings integrated into the AML (Faria *et al.*, 2014a) ontology matching system. This algorithm exploits AML's *Word Lexicon*, the set of all words in an ontology's vocabulary to which is assigned an EC, reflecting the usage of the word within the ontology.

In a first step, the algorithm performs a pairwise mapping of the labels of the source ontology (O_s) with the labels of the target 1 ontology (O_{t1}). The similarity is calculated by finding the ratio of the sum of the EC of the words shared by the source label (l_s) and the target 1 label (l_{t1}), and the sum of the EC of the words in l_{t1}.

$$sim(l_s, l_{t1}) = \frac{\sum EC(word \in (l_s \cap l_{t1}))}{\sum EC(word \in l_{t1})} \tag{5.1}$$

In an intermediate step the algorithm:

1. filters out all mappings with similarity below a given threshold, chosen by the user;
2. removes all the source classes which were not mapped to any target 1 classes;
3. removes from the source labels of mapped classes all the words that had an exact match with a target 1 word.

This reduces the search space, since only source classes for which a partial mapping was found are then matched to target 2. Taking as an example the mapping in Figure 2.4, after matching HP and FMA, which would capture the mapping for 'aorta', the HP's class label would be reduced to 'stenosis'.

In a second step, for each mapping, a pairwise comparison of the reduced source labels with target 2 labels is performed. However, here the ratio divisor corresponds to the sum of EC of the words in the label with more words, to ensure the longest possible match.

$$sim(l_s, l_{t2}) = \frac{\sum EC(word \in (l_{s*} \cap l_{t2}))}{\sum EC(word \in longest(l_s, l_{t2}))} \tag{5.2}$$

The final similarity between the matched labels is computed as the average between the similarities computed in steps 1 and 2. Label mappings below the second threshold are filtered out.
Pseudocode for the algorithm is supplied in Appendix B.

5.1.2 Experiments

Several experiments with external algorithms were performed to try to improve the base compound matching algorithm.

5.1.2.1 Stemming

Due to the nature of the algorithm presented, it is impossible to find matches between words that have small modifications like “cell” and “cells” if those synonyms are not directly expressed in the ontology. Therefore I used stemming, a technique that reduces words to index terms. I employed a popular stemmer implementation, the Snowball stemmer (Porter, Martin F, 2001), which was applied to AML’s Lexicon class.

To use the stemmed words for alignment, each label of the ontology is split in its various words, the stemmer is applied to each word and the main data structure stores the reassembled class label with each word reduced to its stem.

For instance, when applying the Snowball stemmer to the Mammilian Phenotype Ontology (MP) class ’absent respiratory mucosa goblet cells" (MP:0010863) the lexicon saves the label as “absent respiratori mucosa goblet cell". Thus part of the label will fully match with “respiratory goblet cell" (CL:0002370), since “cells” was reduced to “cell”.

5.1.2.2 String Matcher Extender

The String Matcher was used as a secondary matcher after the first alignment to extend the alignment by matching parents, children and siblings of every class mapped in the first alignment using string similarity techniques. This technique is able to expand the mappings in the alignment by adding new ones such as “abnormal granulocyte physiology” (MP:0002462) with “granulocyte” (CL:0000084). This mapping was extended from the mapping “abnormal leukocyte physiology”

(MP:0002442) with "myeloid leukocyte" (CL:0000766), which are the parents of the source and target classes the new mapping. This matcher takes $O(N^2)$ time, which is why it is used as a secondary matcher for a selected number of classes.

5.1.2.3 Thesaurus

In order to extend the lexicon and find more synonyms, which were not directly expressed in the ontology, I used a method which is implemented in AML, called Thesaurus (Pesquita *et al.*, 2013). This extends the lexicon by exploring the relationships established between ontology terms and from there derives new synonyms.

For example, "abnormal vision" (MP:0002090) has the synonym "abnormal visual ability", which supports the inference that the terms "visual" and "visual ability' are synonyms. These synonyms terms are then used to create novel synonyms, by substituting terms with their synonyms in existing names.

5.1.2.4 WordNet

AML's use of WordNet extends the lexicon to find synonyms and hypernyms and generates new labels by replacing the original word with their WordNet synonyms. I applied AML's WordNet class directly to the ontologies to extend their lexicon.

5.1.3 Selection Algorithms

The selection algorithm is used to trim a Compound Alignment by excluding competing mappings (i.e., multiple mappings that include the same source class) to obtain the desired cardinality.

Based on AML's Ranked Selector, I developed two ranked selection algorithms to be applied to compound alignments:

1. **Strict selection.** This greedy selection algorithm selects mappings based on their similarity. It starts by sorting the mappings in the compound alignment in descending order of their similarity values and, if there are competing mappings, selects the one with the highest similarity to be added

to the final alignment. This selection algorithm returns an alignment with a one-to-one cardinality.

2. **Permissive selection.** This algorithm is similar to the strict selection, since it starts by sorting the alignment in a descending order of similarity. However, if competing mappings exist, instead of choosing the one with the higher similarity, it saves the two mappings with the highest similarity in the final alignment. This selection algorithm returns an alignment with a maximum cardinality of one-to-two.

5.1.4 AML adaptation and extension

The algorithms above were implemented in AML. In its original form, AML is an ontology matching system that creates alignments between two ontologies and thus all of its structures and objects were created to handle only two ontologies at a time. Therefore, before integrating the compound matching and selection algorithms in AML's framework it was necessary to adapt all of the necessary structures to be able to handle three different sources of information. For that, I started by adapting the most basic utilities, which include several HashTables that are the base elements to save mappings, and modified them to allow the addition of a third variable. Likewise, a Compound Mapping object was also created, based on the original Mapping structure, but was adapted to handle three classes in one mapping.

AML was also extended to be able to open three ontologies simultaneously and save the three corresponding Lexicon structures. The Word Lexicon structure (similar to the Lexicon, but saves each word of the label individually) was also modified to be able to save numbers and words with less than three characters.

Finally, I adapted the Alignment class and its functions to create compound alignments, save compound mappings instead of regular mappings and to be able to correctly retrieve their information. The evaluation process is done within this class and so I also adapted this process to compare compound alignments against compound reference alignments.

5.2 Evaluation

The evaluation process began by choosing a standard threshold to use throughout the performed tests. Two types of evaluation were performed: automatic and manual. The first compares the obtained alignment to a reference and the second is a manual analysis of all the mappings.

5.2.1 Thresholds

The thresholds used throughout the evaluation process were defined through a series of tests aimed to find consistent values across all 6 sets of ontologies, which returned the best statistics. The tests were performed in all ontology sets.

The first step of the evaluation process involved testing the algorithms with different thresholds for the two matching steps and checking which were the two optimal values to use throughout the evaluation process. These values had to return good statistical values, but also needed to allow the matching process to run in a reasonable amount of time, with a considerable amount of mappings found.

For the first matching step the algorithm needs to return a higher recall, in order to not narrow the search space too much, while still providing enough filtering capabilities within reasonable time. For this to happen the first threshold needs to be lower to increase the recall, at expense of the precision. To determine the best first threshold the source to target 1 mappings were evaluated against the reference alignments in terms of the recall percentage and the total number of mappings found. The logic behind this is to ensure good recall while minimizing the number of source classes to match in the second step, thus reducing the search space. The second optimal threshold was the one that when combined with the first threshold returned the best F-Measure, within a reasonable amount of time.

5.2.2 Automatic Evaluation

To evaluate the algorithms a set of six reference alignments (Pesquita *et al.*, 2014) automatically created by inferring compound mappings from logical definitions (Mungall *et al.*, 2011) in OBO ontologies (Smith *et al.*, 2007) was used and was

compared to the alignments obtained from the algorithms. This approach allowed the use of precision, recall and f-measure to evaluate the results.

These metrics are based on the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) identified when comparing two alignments. In this case, a true positive is a mapping which is present in both the reference alignment and the alignment obtained from the algorithms, whereas a false positive is a mapping identified by the algorithms, which is absent from the reference alignments. A false negative is a mapping in the reference not present in the algorithm's alignment and a true negative is a mapping that is not present in the reference or the obtained alignment. The precision, recall and f-measure are calculated in the following ways:

$$Precision = \frac{TP}{TP + FP} \tag{5.3}$$

$$Recall = \frac{TP}{TP + FN} \tag{5.4}$$

$$F - measure = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{5.5}$$

In this case, the precision represents the number of mappings in the alignment obtained through the algorithms which are correct. The recall represents the fraction of mappings of the reference which are represented in the obtained alignment. Finally, the f-measure is the harmonic mean between the precision and recall.

5.2.3 Manual Evaluation

I performed a manual evaluation of the alignments obtained, using the strict ranked selector to maximize the precision and to reduce the number of mappings returned by the algorithm. The mappings were classified into three possible categories:

- 'Correct', where the mapping is deemed correct and the source class has no competing mapping in the reference alignment;

- 'Conflict', where the mapping is considered potentially correct but there is a competing mapping in the reference alignment using different target classes;

- 'Incorrect', where the mapping is deemed incorrect.

To better understand the 'conflict' mappings, I analysed them by verifying class definitions and ancestry and classified each mapped source as being better matched in the reference alignment or the resulting alignment.

Finally, I built a new set of reference alignments from the original ones, but added the mappings deemed 'Correct' and altered the conflict mappings to the ones that were evaluated as more correct. Therefore, with this extended set of reference alignments it was possible to verify how the new mappings could complement the reference alignments and improve the understanding of the performance of the algorithms.

5.3 Results and Discussion

This section covers the results and discussion of the threshold tests, the automatic and manual evaluations and the outcome of the different experiments.

5.3.1 Thresholds

Table 5.1 presents the results for the first alignment step thresholds experiments ranging from 0.1 to 0.6, using the MP-GO alignment. This range was selected since thresholds above 0.6 are commonly used thresholds for full equivalence binary mappings. The tests performed in the remaining ontologies to find the first threshold are presented in Appendix C, Table C.1 and show the same results as the ones discussed for the MP-GO set.

Table 5.1: Tests of different first thresholds for the MP-GO set.

Threshold	Precision	Recall	F-Measure	Mappings	Running Time
0.1	N/A	N/A	N/A	N/A	$>$ 15h
0.2	0.0 %	45.0 %	0.1 %	1396136	$\approx$ 17min
0.3	0.1 %	43.0 %	0.2 %	376583	$\approx$ 2min
0.4	0.3 %	40.4 %	0.6 %	120025	15s
0.5	0.8 %	39.1 %	1.6 %	44598	7s
0.6	2.0 %	36.5 %	3.7 %	17670	6s

Table 5.1 shows that after 15 hours running time, the alignment with the 0.1 threshold was still not completed. The 0.2 threshold obtained the highest recall (45.0%) with the highest number of mappings found, but also the highest running time from all the threshold experiments that were completed. The 0.3 threshold still returns a large amount of mappings, despite the more reasonable running time. Thus, thresholds below 0.4 have long running times and return alignments that are potentially too large to be efficiently used as filtered input for the second step. For larger alignments, like HP-FMA-PATO, this would be even more of a challenge. Due to these issues the 0.1, 0.2 and 0.3 thresholds were not used when testing with the remaining sets of ontologies (see Appendix C, Table C.1).

The 0.4 and 0.5 threshold obtained similar recalls (40.4 and 39.1%, respectively), with close running times, but the 0.5 threshold obtained almost a third less mappings than the number obtained with the 0.4 threshold, with a small and almost negligible reduction to the recall value.

The 0.6 threshold returned the lowest recall (36.5%) and number of mappings. This value was discarded since it did not bring any improvements to the results.

This evaluation was followed by the tests for the second threshold. In these tests, the first threshold used was either 0.4, 0.5 or 0.6, and the second thresholds ranged from 0.7 to 0.9. Table 5.2 shows these results for the MP-GO-PATO set, but the discussion of the results for the remaining ontologies sets is similar and can be found in Appendix C, Table C.2.

Table 5.2: Tests of different second thresholds for the MP-GO-PATO set.

Thresh 1	Thresh 2	Precision	Recall	F-Measure	Mappings	Time
0.4	0.7	17.9 %	65.1 %	28.1 %	3438	≈ 8min
0.4	0.8	29.3 %	64.3 %	40.2 %	2073	≈ 8min
0.4	0.9	62.8 %	60.6 %	61.7 %	911	≈ 8min
0.5	0.7	19.6 %	63.5 %	30.0 %	3049	≈ 3min
0.5	0.8	30.9 %	62.8 %	41.5 %	1917	≈ 3min
0.5	0.9	63.4 %	59.2 %	61.2 %	882	≈ 3min

Table 5.2 shows that the lowest F-Measure is obtained with 0.4/0.7 thresholds (28.1%), but this pair returns the higher recall of all the pairs (65.1%). However, both the 0.7 and 0.8 thresholds perform significantly worse with either of the threshold 1 values.

The 0.4/0.9 pair has the higher F-Measure (61.7%), but runs for about 8 minutes. However, the 0.5/0.9 pair returns a close F-Measure (61.2%), with a higher precision (63.4% against 62.8% in the 0.4/0.9 pair) and a lower running time (approximately 3 minutes).

With a lower first threshold, more classes are mapped in the second step, which results in a wider search space for the second step. However, the goal here was find the optimal thresholds, which returned the best statistics within a reasonable amount of time. While reducing the first threshold increased the overall performance, especially the recall percentage, it reduces the final precision. Therefore, choosing the best thresholds is a matter of defining if the main goal of the alignment is to maximize the precision or the recall.

5.3.2 Automatic evaluation

The automatic evaluation is defined by testing alignments, obtained with different settings, against automatically generated reference alignments. Here the aim was to evaluate the performance of the algorithms developed and testing them with the application of different methods to improve those results. The thresholds chosen to run these tests were 0.4 for the first matching step and a 0.9 threshold

for the second matching step, since these were the thresholds which achieved a better statistical performance in a reasonable amount of time.

5.3.2.1 Compound Matching Algorithm

Table 5.3 uses the compound matching algorithm and compares each of the alignments against a reference. This table shows the precision, recall, F-Measure, total number of mappings found by the algorithm and how many of them are present in the references. It also presents the size of the reference for each of the alignments.

Table 5.3: Evaluation results against reference alignments using the compound matching algorithm.

Ontology sets	Precision	Recall	F-Measure	Mappings	Correct	Reference
MP-CL-PATO	30.0 %	33.3 %	31.6 %	527	158	474
MP-GO-PATO	42.3 %	58.5 %	49.1 %	1304	552	944
MP-NBO-PATO	40.8 %	35.6 %	38.0 %	191	78	219
MP-UBERON-PATO	42.9 %	32.0 %	36.7 %	1493	640	1999
WBP-GO-PATO	11.4 %	12.3 %	11.8 %	351	40	325
HP-FMA-PATO	29.0 %	12.5 %	17.5 %	818	237	1893

Table 5.3 shows that the compound matching algorithm returns low precision values, with the highest belonging to the MP-UBERON-PATO set (42.9%). The recall values are also quite low, with only the MP-GO-PATO surpassing the 50% mark, with 58.5% recall. Thus, these results illustrate that the compound matching algorithm struggles to deliver good performance when used without selection or other extensions.

The low precision percentage found for the HP-FMA-PATO is mainly due to several classes being wrongly matched as a result of their order being changed, despite both sharing the same words. For instance, the alignment returned the mapping "abnormality of the epiphysis of the distal phalanx of the thumb" (HP:0009662) with "distal epiphysis of phalanx of thumb" (FMA:40297) and "abnormal" (PATO:0000460). Despite, the source and target 1 label sharing all the relevant words, they are present in an order which modifies their meaning, making the mapping wrong. This was one of the patterns discovered in Chapter 4

and HP-FMA-PATO has the highest incidence of these cases, which drastically reduces the precision of this alignment.

5.3.2.2 Experiments

The first experiment consisted in the application of a stemmer to the lexicon of the ontology. The results of this experiment are presented in Table 5.4.

Table 5.4: Automatic evaluation with the application of a stemmer.

Ontology sets	Precision	Recall	F-Measure	Mappings	Correct	Reference
MP-CL-PATO	26.3 %	40.5 %	31.9 %	729	192	474
MP-GO-PATO	40.3 %	60.5 %	48.4 %	1417	571	944
MP-NBO-PATO	42.5 %	41.1 %	41.8 %	212	90	219
MP-UBERON-PATO	40.9 %	39.9 %	40.4 %	1951	797	1999
WBP-GO-PATO	11.5 %	13.5 %	12.4 %	382	44	325
HP-FMA-PATO	23.7 %	22.9 %	23.3 %	1826	433	1893

Results from table 5.4 show a small improvement over those in table 5.3, with some decline in the precision and an increase in recall. The MP-GO-PATO set is the only one which returns a lower f-measure, since it has both a lower precision and recall. However, the application of the stemmer returned a higher number of mappings. For instance, classes which only differed in the terms “cell” and “cells” could be aligned. Yet, the stemmer also introduces some erroneous mappings like ’chronic sinusitis’ (HP:0011109) with ’sinus’ (FMA:51235) and ’chronic’ (PATO:0001863). Since the word ’sinusitis’ was stemmed to ’sinus’, the algorithm gave this mapping the maximum confidence possible, when the mapping is not correct. The application of the stemmer is an optional parameter in the proposed algorithm, and its use depends on the ontologies to align.

The second experiment consisted in the extension of the alignment with AML’s String Matcher, after the first matching step. However, this experiment did not improve the results for any of the ontology sets.

The Thesaurus and WordNet experiments did not yield any results since the matching process became impossible to compute because of RAM memory constraints, due to the extreme increase in size of the Lexicon of the ontologies.

5.3.2.3 Selectors

When applying the matching algorithm by itself, the resulting alignment is a list of all mappings above the selected threshold, without any consideration for cardinality. For instance, if for a given source class there are five mappings scored 0.9, 0.9, 0.91, 0.95 and 0.99, all would be present in the final alignment. To ensure a proper cardinality, selection strategies need to be employed. The reference alignments have a cardinality of 1, meaning that for each source class there is a single compound mapping. However, given the potential for conflicts, it was also desirable to investigate the option of allowing two mappings for the same source class. To this end I developed both a strict and a permissive ranked selector.

Table 5.5 uses the strict ranked selector on the alignment obtained with the compound matching algorithm, with the stemmer applied to the lexicon of the ontologies, since it improved the results of the majority of the ontology sets.

Table 5.5: Evaluation results from the comparison with the automatically generated reference alignments with the Strict Ranked Selector.

Ontology sets	Precision	Recall	F-Measure	Mappings	Correct	Reference
MP-CL-PATO	24.5 %	24.3 %	24.4 %	470	115	474
MP-GO-PATO	62.9 %	60.7 %	61.8 %	911	573	944
MP-NBO-PATO	50.0 %	39.7 %	44.3 %	174	87	219
MP-UBERON-PATO	55.2 %	46.8 %	50.7 %	1693	935	1999
WBP-GO-PATO	11.7 %	10.2 %	10.9 %	283	33	325
HP-FMA-PATO	27.3 %	20.3 %	23.3 %	1409	384	1893

Table 5.5 shows that the highest F-measure is 61.8% for MP-GO-PATO and the lowest 10.9% for HP-FMA-PATO. The precision is consistently higher than the recall, as expected from a selector of this nature. However, despite the improvement over the results with no selector, the algorithm still has a low performance, with only three sets achieving a precision over 50% and only one with a recall over this mark.

Table 5.6 represents the same evaluation process as table 5.5, but using the permissive ranked selector instead of the strict ranked selector.

Table 5.6: Evaluation results from the comparison with the automatically generated reference alignments with the Permissive Ranked Selector.

Ontology sets	Precision	Recall	F-Measure	Mappings	Correct	Reference
MP-CL-PATO	34.9 %	53.0 %	42.0 %	720	251	474
MP-GO-PATO	41.5 %	61.5 %	49.6 %	1399	581	944
MP-NBO-PATO	42.7 %	41.1 %	41.9 %	211	90	219
MP-UBERON-PATO	52.8 %	51.4 %	52.1 %	1947	1028	1999
WBP-GO-PATO	11.6 %	13.5 %	12.5 %	380	44	325
HP-FMA-PATO	24.0 %	22.9 %	23.4 %	1803	433	1893

Table 5.6 shows that the permissive ranked selector returns higher recall values for all ontology sets, sometimes at the expense of the precision. Overall the recall increased an average 2.2% in all sets, except for the MP-CL-PATO which increased 15% when comparing with the strict ranked selector. The MP-CL-PATO was also the only set which obtained a significantly higher precision with the permissive selector than the strict selector. Four of the six sets of ontologies obtained a higher f-measure due to the marked increase in the recall values.

Both the increase in recall and decrease in precision can be explained by the presence of two mappings for some source classes instead of one. For example, both the mapping "retinal ganglion cell degeneration" (MP:0008067) with "retinal ganglion cell" (CL:0000740) and "degeneration" (PATO:0002037) and the mapping "retinal ganglion cell degeneration" (MP:0008067) with "retinal ganglion cell" (CL:0000740) and "degenerate" (PATO:0000639) are present in the alignment with the permissive selector. However, when using the strict selector, only one of these mappings will be present in the alignment. Since both mappings have the same similarity, the one chosen for the final alignment will be randomly selected and it is not necessarily the one featured in the reference alignment. Thus the presence of both mappings reduces the precision, because one of them is always wrong, but increases the recall, since the alignment covers more mappings from the reference.

The MP-CL-PATO benefits the most from the permissive selector, because it is the set which contains more of these competing mappings. The prevalence of similar mappings points to the possibility of the existence of disagreement between reality models in the source and target ontologies, i.e., sister classes in

one ontology might be considered synonyms in another. For instance, "present in fewer numbers in organism" (PATO:0001997) is a sister class of "has fewer parts of type" (PATO:0002001) and despite having different definitions, both have as exact synonym "decreased number". Thus when matching this ontology to others, these two classes can have the same meaning and lead to the present of competing mappings. These mappings create the need for two-to-one alignments, and a selection algorithm such as the permissive selector will thrive in these kinds of alignments, since it can cover more possibilities than the strict ranked selector.

5.3.3 Manual Evaluation

The manual inspection of mappings was the next step in the evaluation and its goal was to ascertain which mappings were correctly or incorrectly identified. For this evaluation the strict selector and the 0.5/0.9 thresholds were chosen, since they achieve a higher precision and return a lower number of mappings. The results are presented in table 5.7. The "Mappings" column shows the number of mappings found by the algorithm. The "Correct" column defines mappings which were deemed correct and it also shows, in parenthesis, the percentage of those correct mappings which are new, i.e., mappings not present in the reference alignment; the "Conflict" column is used when the mapping is considered potentially correct but there is a competing mapping in the reference alignment using different target classes; and "Incorrect" column is for mapping deemed incorrect.

Table 5.7: Manual evaluation of results.

Ontologies	Mappings	Correct (New)	Conflict	Incorrect
MP-CL-PATO	448	47.1 % (17.6 %)	35.3 %	17.6 %
MP-GO-PATO	875	86.9 % (22.3 %)	9.0 %	4.1 %
MP-NBO-PATO	169	70.4 % (20.7 %)	29.6 %	0.0 %
MP-UBERON-PATO	1413	83.5 % (24.7 %)	13.6 %	2.9 %
WBP-GO-PATO	272	44.9 % (33.3 %)	50.4 %	4.7 %
HP-FMA-PATO	1270	81.5 % (44.1 %)	14.4 %	4.1 %

The manual inspection of the mappings, showed in table 5.7, revealed that the algorithm is finding mostly correct mappings, since almost all of them are

above the 50% mark, with the highest percentage of correct mappings belonging to MP-GO-PATO (86.6%), closely followed by MP-UBERON-PATO (83.5%). Both of these alignments have low conflict (9.0% and 13.6%, respectively) and incorrect mappings (4.1% and 2.9%). Only MP-CL-PATO and WBP-GO-PATO have “Correct" percentages below the 50% mark with 47.1 and 44.9% of mappings deemed correct. However, they also have the highest percentage of conflict mappings (35.3% and 50.4%, respectively).

Table 5.7 also shows that the WBP-GO-PATO returns the highest number of new mappings (33.3%), with all others above the 20%, except for the MP-CL-PATO alignment which only has a 17.6% of correct mappings not present in the reference.

Finally, the percentage of incorrect mappings is low for all alignments, with MP-NBO-PATO being the lower (0%) and the MP-CL-PATO reaching the highest proportion (17.6%). This higher percentage is due to the fact that the CL ontology describes cells, so, many of its classes will have terms which can be mapped to the MP ontology, but instead of describing a tissue, they describe the cell of the tissue. For example, 'absent mesoderm' (MP:0001683) is mapped to 'mesodermal cell' (CL:0000222) and 'lacks all parts of type' (PATO:0002000). There is a high number of cases which fall in this category and are considered incorrect.

5.3.4 Conflict Analysis

To better understand the previous results I expanded the analysis of the conflict mappings by verifying class definitions and ancestry and classified each mapped source as being better matched in the reference alignment or the resulting alignment. For mappings for which there was no clear evidence either way, I decided to consider the reference mapping as the correct mapping.

Table 5.8: Manual analysis of the conflict mappings showing the percentage of mappings more correct in the alignments obtained with the algorithms.

Ontologies	Number of Conflicts	Correctly Aligned
MP-CL-PATO	158	7.6 %
MP-GO-PATO	79	65.8 %
MP-NBO-PATO	50	74.0 %
MP-UBERON-PATO	192	42.7 %
WBP-GO-PATO	139	86.3 %
HP-FMA-PATO	185	53.5 %

Table 5.8 shows that most of the conflict mappings found by the algorithms could potentially be considered more accurate than those currently present in the reference alignments. For example, 'flattened snout' (MP:0000447) with 'snout' (UBERON:0006333) and 'flattened' (PATO:0002254) could be considered more accurate than the mapping present in the reference alignment, which is 'flattened snout' (MP:0000447) with 'midface' (UBERON:0004089) and 'flattened' (PATO:0002254).

This conflict analysis revealed a number of causes for these conflicts:

1. **The definition specifies the class.** Mainly in the HP ontology I found several cases where the source class specified which FMA class was used to build that particular class. For example, 'Absent ethmoidal sinuses' is defined as 'Lack (aplasia) of the 'ethmoidal sinus' (FMA:84115)'. Most of these cases were wrongly identified in the reference alignments, but were correctly matched by the algorithm. Thus, in all of these cases I chose the mapping which figured the FMA class present in the definition.

2. **Mappings using the alternate id.** I found several instances where the URI present in the reference alignment was the "alt_id", instead of the main identification number. For example, the mapping 'deep philtrum' (HP:0002002) with 'philtrum' (FMA:59819) and 'increased depth' (PATO:0001596) is present in the reference alignment, but the PATO class is identified as PATO:0001666, which is the alternate id for 'increased depth'. Currently, AML evaluates the alignments using only the main id, therefore,

those mappings counted as incorrect, when in fact they were correct. In those cases I chose to change the reference alignment to have the main id instead of the alternate id.

3. **Inconsistencies when matching classes.** The reference alignment has mappings such as 'Hypoplastic tibia' (HP:0005736) with 'Bone of tibia' (FMA:33837) and 'hypoplastic' (PATO:0000645), but when the tibia appears in another mapping, it is a class from a different branch, for no apparent reason. For example, 'Short tibia' (HP:0002993) with 'Tibia' (FMA:24476) and 'decreased length' (PATO:000574). By looking at the ancestry of both FMA classes I decided the correct term was 'Tibia', because 'Bone of tibia' originated from the class 'Portion of tissue' (FMA:9637) and 'Tibia' belonged to the 'Organ' (FMA:67498) class. Therefore, despite being somewhat subjective, I decided that it was more accurate to align all of the instances similar to these with the classes from the 'Organ' branch since it represents the whole bone and not just a portion of a tissue. I followed the same strategy for all similar cases, addressing each individually and decided which was better suited for each particular case.

4. **Mappings using an incorrect label.** There were cases where the reference alignment featured a synonym as label, instead of the main label. For example, 'Absent tibia' (HP:0009556) is present in the reference alignment as 'Aplasia of the tibia', an exact synonym of the class. So, in this case, the reference alignment shows 'Aplasia of the tibia' (HP:0009556) with 'Tibia' (FMA:24476) and 'aplastic' (PATO:0001483) and the algorithm aligns the PATO class with 'lacks all parts of type' (PATO:0002000) instead. Here I decided to consider correct the mapping which aligned more accurately with the label of the class in the source ontology.

5. **Class specificity.** When a conflict had one of the target classes differing only by a level in the ontology, i.e., the reference had a parent and the alignment the child, or vice-versa, I always chose the more specific one, which is always the child. So, for example, in the mapping 'intervertebral disc degeneration' (HP:0008419) with 'intervertebral disk' (FMA:10446) and 'de-

generation' (PATO:0002037) present in the obtained alignment, I chose to keep the mapping present in the reference, because instead of 'degeneration', the source class was mapped to 'degenerate' (PATO:0000639), which is one level lower than 'degeneration' and therefore more specific.

6. **Disambiguations.** When I found class labels which were synonyms, I chose to keep the target which was more closely related to the source label. For example, 'flattened snout' (MP:0000447) with 'snout' (UBERON:0006333) and 'flattened' (MP:0002254) was in conflict with the same mapping in the reference, but changed the target 1 to the synonym 'midface' (UBERON:0004089). In all instances of these types of cases I kept the exact label match instead of the synonym.

7. **Wrong mappings.** There were several instances where the conflict mapping was simply wrong in one of the alignments and in all of those cases I was able to consider as correct the mapping in the other alignment. For example, the algorithm found 'abnormal innervation' (MP:0002184) with 'neuron projection' (GO:0043005) and 'abnormal' (PATO:0000460). This mapping is present in the reference with the source class mapped to 'innervation' (GO:0060384) which would be the correct match. However, the reference also has some incorrect conflict mappings like 'abnormal menstrual cycle' (MP:0003375) with 'menstruation' (GO:0042703) and 'abnormal' (PATO:0000460), which is correctly aligned by the algorithm to 'menstrual cycle' (GO:0044850) instead.

WBP-GO-PATO has the highest percentage of potentially more correct conflict mappings (86.3%) obtained through the proposed algorithm and MP-CL-PATO has the lowest (7.6%), which means that most of the conflict mappings are correct in the reference alignment for this set. The majority of the other alignments are above or relatively close to the 50% mark.

To explain the low number of correct conflicts in the MP-CL-PATO I can refer to the disambiguation issue, because this alignment has a high number of conflict mappings which involve a reference to an increased/decreased number of cells. The algorithm always matches those cases to 'decreased amount'

(PATO:0001997)/'increased amount' (PATO:0000470) instead of 'has fewer parts of type'(PATO:0002001)/'has extra parts of' (PATO:0002002), which is present in the reference alignment. About 140 of the 158 MP-CL-PATO conflicts have this problem and in all of them I decided to go with the reference, because the definition of 'has fewer parts of type' fits better the description of the source class, which is almost always an increased/decreased number of a certain cell.

The high number of correctly aligned conflicts in WBP-GO-PATO can be explained by the misuse of labels to align. Several source classes of the reference of this set have 'abnormal' in the label, instead of the preferred name which uses the term 'variant'. For example, 'cell development abnormal' (WBPhenotype:0000529) should be 'cell development variant' (WBPhenotype:0000529). So reference aligns with 'abnormal' (PATO:0000460), instead of 'variant' (PATO:0001227). In these cases I chose to keep the match the algorithm makes, which is always with 'variant'. This decision accounts for almost all of the correct conflicts in the WBP-GO-PATO alignment.

Following this analysis, the reference alignments were extended with the new correct mappings and conflicting mappings which were deemed correct in the alignment obtained through the algorithms. Then I ran the algorithms again, using the strict selector, against this new set of reference alignments and those results are presented in table 5.9.

Table 5.9: Evaluation of the corrected reference alignments.

Ontologies	Precision	Recall	F-Measure	Mappings	Correct	Reference
MP-CL-PATO	43.2 %	36.7 %	39.7 %	470	203	553
MP-GO-PATO	89.8 %	71.9 %	79.8 %	911	818	1138
MP-UBERON-PATO	79.9 %	57.6 %	67.0 %	1693	1353	2348
WBP-GO-PATO	83.7 %	57.0 %	67.8 %	283	237	416
HP-FMA-PATO	84.5 %	48.4 %	61.6 %	1409	1190	2457
MP-NBO-PATO	91.4 %	63.1 %	74.6 %	174	159	252

Table 5.9 shows, as expected, that most alignments would fall above de 50% mark for F-Measure. However, despite the fact that almost all the sets achieve a precision close to 90%, the MP-CL-PATO set is still below 50%. Since this set had a low percentage of new mappings discovered (see Table 5.7) and a high

percentage of wrong conflict mappings (see Table 5.8), the number of new mappings added to the expanded reference was low. This observation confirms the fact that this ontology set has a significant amount of classes which the algorithm struggles to differentiate between (as referenced in section 5.3.2.3).

The precision was not the only result affected, the recall slightly increased in all sets, however not as much as the precision. Thus the algorithm is still missing a considerable amount of mappings which are present in the reference alignment. This is mainly due to mappings in the references which are semantically related through synonyms not explicitly expressed in the lexicon of the ontology, such as 'polyuria' (MP:0001762) mapped to 'micturition' (GO:0060073) and 'increased rate' (PATO:0000912). The MP class labelled "polyuria" has three exact synonyms: "increased urine output", "increased urine volume" and "diuresis". None of these synonyms reference the term "micturition" and so the algorithm cannot find this mapping.

5.4 Discussion

One challenge of computing compound alignments is the memory requirements involved in the process. If matching two large biomedical ontologies is already a challenge for many ontology matching systems, handling three ontologies in a compound alignment scenario is even more demanding. The algorithms here presented decrease the search-space by using a two-step matching approach, which both reduces the time and memory requirements[1]. The algorithms handled quite large ontologies in a reasonable amount of time and returned good results.

The algorithms were tested automatically and manually. While the automatic results against the reference alignments underperformed, with lower than expected statistics, the manual evaluation showed that, in the majority of cases, the algorithms were returning mostly correct mappings. This indicated that on one hand, the reference alignments are incomplete, and on the other, that the algorithms are failing to capture a considerable portion of reference mappings. Moreover, this evaluation also led us to finding several mappings which conflicted

[1]The largest alignment, HP-FMA-PATO, takes less than 15 minutes with an Intel® Core™i7-2600 CPU 3.40GHz x 8 processor and 16GB memory.

with the mappings found in the reference alignment. Using the permissive ranked selector, allowed some of the alignments to overcome the issue by featuring both the conflict and the non-conflicting mappings in the alignments at the expense of the precision. This selection strategy can be used in a user interaction scenario, where the user then decides between the two conflicting mappings.

Despite the overall good performance of the proposed algorithms there are still some open challenges:

1. Theoretically, the algorithm is symmetric, i.e., it does not matter which of the target ontologies is the target 1 or 2 to yield results. However, we empirically found that the algorithm performs better if the ontologies are aligned in a specific order. In this case, we always matched PATO last since we consistently obtained better results with this specific order. In the future, it would be desirable to automate the selection order by evaluating the coverage of each of the matching orders.

2. Currently, the user needs to possess specific previous knowledge of the ontologies to be able to perform the alignment, i.e., the user needs to know which two ontologies are able to form a set of terms which is equivalent to the label of the source ontology. One solution to this challenge could be to use several ontologies as input to automatically determine the ontologies which could form potential compound mappings. AML currently uses a similar strategy to determine which ontologies can be used as background knowledge in a binary alignment setting, which can in principle be adapted to selecting the appropriate ontologies for compound matching (Faria *et al.*, 2014b).

3. Exploring external knowledge (such as WordNet) with existing techniques proved to be too computationally demanding. However, given the fact that some mappings need external knowledge (such as synonyms) to be identified, there is still a need to adapt these strategies to a ternary compound matching setting.

Despite the increased challenge of producing ternary compound alignments, the matching strategy returns considerably good results and thus successfully

answers Research Question 2, which asked if it was possible to adapt previous algorithms to the compound matching problem. Although handling the search space was a challenge, Research Question 3 was also solved, since the application of partial label matching allowed the reduction of the search space and the algorithm was able to return an alignment in a reasonable amount of time, even for large biomedical ontologies.

Chapter 6

Practical Applications

This chapter presents two different practical applications for the compound matching algorithms.

6.1 Logical Definitions

The results from the manual evaluation (presented in Table 5.7) led me to investigate how the algorithms could impact the current state of the OBO logical definitions. The results show a significant percentage of new mappings and, since the reference alignments were created from the logical definitions, the next logical step would be to verify if these new mappings could be applied to the maintenance process of the logical definitions.

To test the possibility of using the algorithms to add logical definitions I compared the total number of classes in each of the logical definitions to the number of new mappings discovered by the algorithm. This is fundamentally different than analysing the number of new mappings that can be added to the reference alignment, since the mappings in those alignments need to form a ternary relationship, and logical definitions can include any number of classes from any number of ontologies.

For instance, the following logical definition has more than three intersections and so is not present in the reference alignment:

```
id:  MP:0000137 !  abnormal vertebrae morphology
intersection_of:  PATO:0000051 !  morphology
intersection_of:  has_component PATO:0000460 !  abnormal
intersection_of:  inheres_in UBERON:0002412 !  vertebra
```

In this case, if the algorithm finds a mapping that is already present in the logical definitions as a non-ternary logical definition, it is considered a conflict. All the previous conflicts were competing with mappings from the reference alignments and were, thus, logical definitions which formed a ternary relationship. So I decided to also evaluate the impact which all of these conflicts mappings could have in the maintenance of the logical definitions.

6.1.1 Results and Discussion

Table 6.1 compares the number of new and conflicting mappings produced for each of the three ontologies in relation to the total number of OBO classes represented in the logical definitions. It also shows the potential percentage of growth to the logical definitions, if the new mappings were to be added.

Table 6.1: Candidate logical definitions.

Ontology	New Mappings	Conflicts	OBO classes	% of Growth
MP	335	442	7694	6.5
WBP	72	140	957	7.5
HP	498	169	14059	3.5

Table 6.1 shows that the MP logical definitions could have a potential growth of 6.5%, with 335 new definitions. The WBP could have 72 new candidate logical definitions and the HP ontology could grow in 3.5%, with 498 new logical definitions. This represents more than 900 new candidate logical definitions for classes that had none.

However, the algorithms also produce more than 750 mappings that are in conflict with the logical definitions. Over 400 of these correspond to non-ternary

logical definitions. For instance, the logical definitions of the MP ontology contains the following:

```
id:  MP:0004403 !  absent cochlear outer hair cells
intersection_of:  PATO:0002000 !  lacks all parts of type
intersection_of:  inheres_in UBERON:0001844 !  cochlea
intersection_of:  towards CL:0000601 !  outer hair cell
```

This logical definition is present in the alignment as "absent cochlear outer hair cells" (MP:0004403) with "outer hair cell" (CL:0000601) and "lacks all parts of type" (PATO:0002000). This mapping is not erroneous, because "cochlear outer hair cells" is an exact synonym of this label. However, because a logical definition for this source class existed, this mapping was added to the number of conflicts. Cases such as these showcase the possibility of producing more than one correct logical definition for each class.

Besides these non-ternary conflicts, the ternary conflicts still account for more than 300 possible new logical definitions. Therefore, an expert analysis total of the conflict mappings can potential reveal novel logical definitions that can further improve the ontologies.

Both the addition of new mappings and the use of the conflict mappings could potentially lead to the improvement of logical definitions. This method could go beyond the currently employed methods, like Obol, since it can find new potential logical definitions, and propose improved alternatives to some pre-existing ones. However, validation by expert curators would still remain necessary to ensure correctness.

6.2 Crop Ontology

The "Planteome" project, funded by the American National Science Foundation, aims to build, revise and promote the use of a set of ontologies of reference for plants. In the context of that project, one of the tasks involves aligning crop the specific Wheat Crop Ontology (CO) (Shrestha *et al.*, 2010) to reference ontologies (Plant Trait Ontology (TO)) (Arnaud *et al.*, 2012),Plant Ontology (PO) (Avraham *et al.*, 2008) and PATO). Their initial plan was to use the standard

AML matchers to complete the task. However, they needed to find more complex matches like "leaf length" (CO:321_0000044) with "leaf" (PO:0025034) and the PATO class "length" (PATO:0000122). The purpose of this work is to create something similar to OBO logical definitions for plant traits, i.e. creating formal definitions, which would allow reasoning over these ontologies. So it was proposed the use of these new compound algorithms to obtain those matches.

6.2.1 Methods

The ontologies used are shown in Table 6.2 and they were divided into two sets: CO-PO-PATO and TO-PO-PATO. To find the optimal thresholds for this new data, I performed similar tests to the ones presented in section 5.2.1, but since there is no reference alignment available, I manually checked each of the alignments and chose the thresholds which resulted in a reasonable amount of mappings found, with significant correct results.

The weight attributed to narrow synonyms also needed to be adjusted, since these ontologies predominantly use these synonyms, unlike the biomedical ontologies used, which rarely use them.

Table 6.2 is similar to Table 3.1, but presents the number of classes and names of these plant related ontologies.

Table 6.2: Plant ontologies.

Ontology	Acronym	Classes	Names	Reference
Wheat Crop Ontology	CO	240	1340	Shrestha *et al.* (2012)
Plant Trait Ontology	TO	1337	5312	Arnaud *et al.* (2012)
Plant Ontology	PO	1691	15544	Avraham *et al.* (2008)

Table 6.2 shows that comparatively to the biomedical ontologies in Table 3.1, these ontologies are small, with the highest number of names (15544) belonging to the PO.

6.2.2 Results and Discussion

Table 6.3 shows a representative selection of the most promising pairs of thresholds with the results regarding the two sets of ontologies tested. It presents the number of mappings found and the percentage which I considered correct.

Table 6.3: Evaluation of the plant based alignments

Threshold 1	Threshold 2	CO-PO-PATO			TO-PO-PATO		
		Found	Correct	Time	Found	Correct	Time
0.1	0.9	14	93 %	20s	259	96 %	149s
0.1	0.7	45	36 %	20s	487	55 %	169s
0.3	0.85	4	100 %	6s	152	95 %	15s
0.5	0.9	0	0 %	5s	25	92 %	7s

Table 6.3 shows that for CO-PO-PATO the highest number of correct mappings (100 %) was obtained with 0.3/0.85 thresholds, but it also returned a small number of mappings (4). For TO-PO-PATO the highest percentage of correct mappings was found using 0.1/0.9 as thresholds (96 %), which could also be considered the best thresholds for CO-PO-PATO, since the algorithm finds more mappings (14) and keeps a relatively high percentage of correct mappings (93 %). Unlike the evaluation performed with the biomedical ontologies, here it is possible to use 0.1 as the first threshold and obtain an alignment in a small period of time, because the ontologies being aligned have a lower number of classes and names.

It was positively unexpected the small amount of adjustments needed to make the algorithms optimally run to obtain significant results, since the whole project was designed to work with large biomedical ontologies. It was surprising to find that the only adjustment needed was to give a higher weight to narrow synonyms, which was too low and was severely reducing the final similarity of the alignments.

These experiments illustrate that the proposed algorithms can be generalized to other life sciences ontologies with positive results.

Chapter 7

Conclusions and Future Work

This chapter summarizes the conclusions of this work, discusses some limitations and future work and presents the final remarks.

7.1 Summary

Biomedical ontologies are crucial to support the management and analysis of life sciences data. Classical binary ontology matching techniques have been used to ensure interoperability between ontologies covering the same or closely related domains. However, in the biomedical field the complexity of relations between entities extends beyond the capabilities of current matching systems. Broadening the concept of ontology matching is necessary to handle this, so the main purpose of this work was to develop novel matching algorithms capable of producing ternary compound alignments, relating classes from three distinct ontologies: a source class equivalent to the intersection of two target classes. The novel compound matching algorithms developed address the specific challenge of not only matching large biomedical ontologies, but also handling the increased matching space and the inherently more difficult-to-compute ternary matching.

This work was divided into three main tasks which aimed to devise, develop, evaluate and apply the compound matching algorithms. The first task consisted in manually analysing binary mappings of biomedical ontologies and their reference alignments. This resulted in the emergence of five distinct label patterns: addition, variation, combination, full match and synonyms. The "addition" led to

the idea of using a bag-of-words approach and "variation" and "synonyms" pattern suggested that several external resources could be used to improve the matching process.

These results were use to devise the strategy for the development of compound matching algorithms in the second task. This task consisted in creating novel compound matching algorithms and evaluating their performance. Since the algorithm would be applied to large ontologies, the main focus during the development of the algorithm was reducing the search space. So a two step approach was devised. The first step consisted in producing an intermediate binary alignment between partial matched labels, removing words and classes not matched and matching the remaining classes and words to the second target. This approach greatly reduces the search space and enables the compound matching of large ontologies. The algorithm finalizes with a selection step to reduce the cardinality to the desired level. Furthermore, integrating these algorithms in ontology matching systems such as AgreementMakerLight, results in a methodology that is particularly suited to match biomedical ontologies, given its ability to handle large ontologies.

The evaluation of the algorithms has shown that, despite the challenges in handling an increased matching space and the inherently more difficult-to-compute ternary mapping, the algorithm is able to produce significant results. Although the evaluation against automated reference alignments produced f-measures between 10.9 and 61.8%, the manual evaluation of these results showed that the algorithms were mostly finding correct mappings, with a significant number of new mappings not present in the reference alignments. However, there was still a significant amount of conflicting mappings (i.e., mappings for the same source class), ranging from 9 to 15% conflicts between the six sets of ontologies. The manual analysis of these conflicts exposed several issues with the reference alignments and the alignments obtained through the algorithms, revealing the complexity of this task.

To further investigate the usefulness of the novel matching algorithms, two applications were tested: the creation of novel candidate logical definitions and the alignment of plant ontologies. Our proposed strategy was able to successfully identify a significant number of novel logical definitions candidates, with a low

error rate. Despite having been specifically developed to handle large biomedical ontologies, the algorithms also proved to be successful in aligning ontologies from the plant domain, need only minor adjustments to produce results.

Despite some standing issues and challenges, the proposed algorithms have proved successful and, to the best of my knowledge, are the first algorithms to be able to produce compound alignments, specifically ternary compound alignments.

7.1.1 Limitations and Future Work

Despite the promising results the algorithms still present some shortcomings and challenges, for which I propose future directions. One such issues is that currently word order is not taken into account by the algorithms, and in same cases two labels which feature the same words in a different order have a different semantic meaning. To handle this the algorithm would need to evolve from a bag-of-words approach to an ordered approach, however the efficiency currently guaranteed by the word structures based on hash maps would need to be maintained.

The proposed algorithm is also not capable of handling matches between classes which have synonyms not explicitly expressed in the lexicon of the ontology. Since the application of external knowledge (such as WordNet) has proven to be too computationally demanding with the current techniques, in the future, exploring the development of new efficient strategies to employ such knowledge could be beneficial to this problem.

Although our current approach is limited to producing new logical definitions with ternary intersections, we expect it can easily be adapted to logical definitions that employ classes from the source ontology and a single external ontology. In the future, it could also be explored how different similarity thresholds can affect the accuracy and coverage of the obtained logical definitions.

Another limitation of these algorithms is their dependence on *a priori* knowledge. Currently, the algorithm takes as input three ontologies, pre-identified as source, target 1 and target 2. This is equivalent to the current scenario of classical binary alignment. However, to increase its applicability it should be possible to automatically identify target order. So far, this is accomplished empirically by running the alignments and comparing them to the reference. Furthermore, it

should also be in principle possible that given a source ontology and a large set of related ontologies, an automated discovery strategy would be able to identify the best pair of ontologies to serve as targets. Therefore, it could be useful to automate the process of discovering the best targets by using several ontologies as background knowledge and, to automate the matching order, the algorithm could determine the best coverage of the two different orders before producing the final alignment. Current strategies for automated discovery of background ontologies for classical matching could be explored in this context.

7.1.2 Final remarks

In this dissertation I addressed the challenges of developing compound matching algorithms to produce ternary compound alignments for biomedical ontologies. Current semantic technologies applied to the biomedical domain are limited to binary mappings, mostly based on equivalence. Although these have proven useful in a number of applications, I believe that the evolving complexity in producing a fruitful analysis of biomedical data will increase the need for more complex semantic relations between entities from different areas. Compound ontology matching can contribute to solving this issue, since having systems which are able to handle this increased complexity will be even more important in the coming years.

It is the successful integration of all the biomedical knowledge that will allow us to continue to advance at a high pace and I believe that the work presented in this dissertation is a meaningful contribution not only in ontology matching but also in the biomedical ontology field and bioinformatics in general.

Appendix A

OWL format class

```
<owl:Class rdf:about="http://purl.obolibrary.org/obo/CL_0000007">
      <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">early
embryonic cell</rdfs:label>
   <rdfs:subClassOf rdf:resource="http://purl.obolibrary.org/obo/CL_0002321"/>
   <obo:IAO_0000115 rdf:datatype="http://www.w3.org/2001/XMLSchema#string">A
cell found in the embryo before the formation of all the gem layers is
complete.</obo:IAO_0000115>
   <oboInOwl:hasOBONamespace rdf:datatype="http://www.w3.org/2001/XMLSchema#string">cell
   </oboInOwl:hasOBONamespace>
</owl:Class>
<owl:Axiom>
   <owl:annotatedTarget rdf:datatype="http://www.w3.org/2001/XMLSchema#string">A
cell found in the embryo before the formation of all the gem layers is
complete.  </owl:annotatedTarget>
   <oboInOwl:hasDbXref rdf:datatype="http://www.w3.org/2001/XMLSchema#string">GOC:tfm
   </oboInOwl:hasDbXref>
   <owl:annotatedSource rdf:resource="http://purl.obolibrary.org/obo/CL_0000007"/>
   <owl:annotatedProperty rdf:resource="http://purl.obolibrary.org/obo/IAO_0000115"/>
</owl:Axiom>
```

Appendix B

Pseudocode for the algorithm

Algorithm 1 Compound Matching Algorithm - Step 1

input: sourceOntology, target1Ontology and target2Ontology

$srcLexicon \leftarrow getSourceLexicon(sourceOntology)$

$tgtLexicon \leftarrow getTarget1Lexicon(target1Ontology)$

$threshold1 \leftarrow n$

$alignment = \emptyset$

$maxSim \leftarrow 0$

foreach $sourceName \in srcLexicon$ **do**
    $weightS \leftarrow getWeight(sourceName)$
    foreach $targetName \in tgtLexicon$ **do**
        $weightT \leftarrow getWeight(targetName)$
        $sim \leftarrow weightS * weightT$
        $sim \leftarrow sim * targetNameSimilarity(sourceName, targetName)$
        if $sim > maxSim$ **then**
            $maxSim = sim$
        end
    end
    if $maxSim > threshold1$ **then**
        $alignment \leftarrow add(classSource, classTarget, maxSim)$
    end
end

B. PSEUDOCODE FOR THE ALGORITHM

```
tgt2Lexicon ← getTarget2Lexicon(target2Ontology)
threshold2 = c
compAlignment = ∅
setMappingWord ← filteringStep(alignment)

maxSim ← 0
foreach setMappingWord ∈ setMappingsWords do
    mapping ← setMappingWord.getMapping()
    words ← setMappingWord.getWords()
    foreach target2 ∈ tgt2Lexicon do
        sim ← nameSimilarity(target2, words)
        if sim > maxSim then
            maxSim = sim
        end
    end
    if maxSim > threshold2 then
        compAlignment ← add(classSource, classTarget1, classTarget2, maxSim)
    end
end
return compAlignment
```

Algorithm 2 targetNameSimilirity - Step 1

```
input: sourcename and target1Name
intersection ← ∅
targetEC ← getNameEC(targetName)
foreach word ∈ sourceName do
    if word ∈ target1Name then
        intersection ← intersection + getEC(word)

    end
end
return intersection/targetEC
```

Algorithm 3 filteringStep

input: alignment

$setMappingsWord = \emptyset$

foreach $mapping \in alignment$ **do**

  $sourceLabel \leftarrow getMappingSourceLabel()$

  $targetLabel \leftarrow getMappingTargetlabel()$

  $newSourceLabel \leftarrow \emptyset$

  foreach $word \in sourceLabel$ **do**

    if $word \notin targetLabel$ **then**

      $newSourceLabel \leftarrow add(word)$

    end

    $setMappingWord \leftarrow add(mapping, newSourceLabel)$

  end

end

return $setMappingWord$

B. PSEUDOCODE FOR THE ALGORITHM

Algorithm 4 nameSimilarity - Step 2

input: sourceWords and target2Name

$intersection \leftarrow \emptyset$

$finalEC \leftarrow \emptyset$

$similarity \leftarrow \emptyset$

if $target2Name.length > sourceWords.length$ **then**
    $finalEC \leftarrow getNameEC(target2Name)$
    foreach $word \in target2Name$ **do**
        if $word \in sourceWords$ **then**
            $intersection \leftarrow intersection + getEC(word)$
        end
    end
    $similarity \leftarrow intersection/finalEC$
else
    foreach $word \in sourceWords$ **do**
        $finalEC \leftarrow finalEC + getEC(word)$
        if $word \in target2Name$ **then**
            $intersection \leftarrow intersection + getEC(word)$
        end
    end
    $similarity \leftarrow intersection/finalEC$
end
return $similarity$

Appendix C

Threshold test results

Table C.1: Tests of different first thresholds for the remaining ontology sets.

Ontologies	Threshold	Precision	Recall	F-Measure	Mappings	Time
MP-CL-PATO	0.4	0.4	20.7 %	0.7 %	26336	5s
	0.5	0.9	19.8 %	1.7%	10865	1s
	0.6	1.6	17.7 %	3.0 %	5158	1s
MP-NBO-PATO	0.4	1.7	26.6 %	3.2 %	3395	1s
	0.5	3.5	21.9 %	6.1 %	1359	0s
	0.6	6.6	19.6 %	9.9 %	649	0s
MP-UBERON-PATO	0.4	1.2	40.2 %	2.4 %	64984	5s
	0.5	2.5	34.1 %	4.6 %	27464	2s
	0.6	4.7	27.0 %	8.0 %	11546	1s
WBP-GO-PATO	0.4	0.8	73.8 %	1.5 %	30501	5s
	0.5	2.1	65.8 %	4.1 %	10090	1s
	0.6	5.0	55.1 %	9.2 %	3575	1s
HP-FMA-PATO	0.4	0.2	69.2 %	0.4 %	552604	133s
	0.5	0.6	55.5 %	1.2 %	176218	20s
	0.6	2.0	43.0 %	3.8 %	40847	9s

C. THRESHOLD TEST RESULTS

Table C.2: Tests of different first (T1) and second (T2) thresholds for the remaining ontology sets.

Ontologies	T1	T2	Precision	Recall	F-Measure	Mappings	Correct	Time
MP-CL-PATO	0.4	0.8	11.3 %	26.4 %	15.8 %	1107	125	121s
	0.4	0.9	24.5 %	24.3 %	24.4 %	470	115	116s
	0.5	0.8	11.4 %	24.9 %	15.7 %	1032	118	48s
	0.5	0.9	24.3 %	23.0 %	23.6 %	448	109	48s
	0.6	0.8	11.5 %	22.6 %	15.2 %	933	107	23s
	0.6	0.9	24.3 %	21.3 %	22.7 %	415	101	23s
MP-NBO-PATO	0.4	0.8	38.8 %	42.9 %	40.8 %	242	94	14s
	0.4	0.9	50.0 %	39.7 %	44.3 %	174	87	12s
	0.5	0.8	39.4 %	41.6 %	40.4 %	231	91	5s
	0.5	0.9	49.7 %	38.4%	43.3 %	169	84	5s
	0.6	0.8	41.4 %	40.6 %	41.0 %	215	89	2s
	0.6	0.9	50.3 %	37.4 %	42.9 %	163	82	2s
MP-UBERON-PATO	0.4	0.8	28.5 %	51.7 %	36.8 %	3625	1034	305s
	0.4	0.9	55.3 %	46.9 %	50.8 %	1693	937	307s
	0.5	0.8	27.9 %	44.9 %	34.4 %	3216	897	124s
	0.5	0.9	58.0 %	41.0 %	48.0 %	1413	819	125s
	0.6	0.8	26.4 %	37.2 %	30.9 %	2815	744	50s
	0.6	0.9	58.4 %	34.2 %	43.1 %	1170	683	51s
WBP-GO-PATO	0.4	0.8	6.9 %	12.9 %	9.0 %	611	42	111s
	0.4	0.9	11.7 %	10.2 %	10.9 %	283	33	106s
	0.5	0.8	7.2 %	12.9 %	9.2 %	586	42	35s
	0.5	0.9	12.0 %	10.2 %	11.0 %	276	33	36s
	0.6	0.8	6.9 %	11.1 %	8.5 %	524	36	13s
	0.6	0.9	11.6 %	8.9 %	10.1 %	251	29	13s
HP-FMA-PATO	0.4	0.8	20.5 %	22.0 %	21.2 %	2025	416	2111s
	0.4	0.9	27.2 %	20.2 %	23.2 %	1409	383	2104s
	0.5	0.8	20.1 %	19.8 %	20.0 %	1862	375	621s
	0.5	0.9	26.9 %	18.3 %	21.8 %	1289	347	620s
	0.6	0.8	18.9 %	16.7 %	17.7 %	1673	316	146s
	0.6	0.9	25.6 %	15.6 %	19.4 %	1152	295	146s

References

Arnaud, E., Cooper, L., Shrestha, R., Menda, N., Nelson, R.T., Matteis, L., Skofic, M., Bastow, R., Jaiswal, P., Mueller, L.A. *et al.* (2012). Towards a reference plant trait ontology for modeling knowledge of plant traits and phenotypes. In *KEOD*, 220–225. 65, 66

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. 1, 8, 24

Avraham, S., Tung, C.W., Ilic, K., Jaiswal, P., Kellogg, E.A., McCouch, S., Pujar, A., Reiser, L., Rhee, S.Y., Sachs, M.M., Schaeffer, M., Stein, L., Stevens, P., Vincent, L., Zapata, F. & Ware, D. (2008). The plant ontology database: a community resource for plant structure and developmental stages controlled vocabulary and annotations. *Nucleic Acids Research*, **36**, D449–54. 65, 66

Bard, J., Rhee, S.Y. & Ashburner, M. (2005). An ontology for cell types. *Genome biology*, **6**, R21. 10, 24

Cheatham, M. & Hitzler, P. (2014). The properties of property alignment. *Ontology Matching*, 13. 5

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. 16

REFERENCES

Day-Richter, J., Harris, M.A., Haendel, M., Lewis, S. *et al.* (2007). Obo-edit—an ontology editor for biologists. *Bioinformatics*, **23**, 2198–2200. xix, 9

Dhamankar, R., Lee, Y., Doan, A., Halevy, A. & Domingos, P. (2004). iMAP: discovering complex semantic matches between database schemas. In *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, 383–394, ACM. 21

Doan, A., Madhavan, J., Dhamankar, R., Domingos, P. & Halevy, A. (2003). Learning to match ontologies on the semantic web. *The VLDB Journal - The International Journal on Very Large Data Bases*, **12**, 303–319. 20

Dragisic, Z., Eckert, K., Euzenat, J., Faria, D., Ferrara, A., Granada, R., Ivanova, V., Jiménez-Ruiz, E., Kempf, A.O., Lambrix, P. *et al.* (2014). Results of the ontology alignment evaluation initiative 2014. In *Proceedings of the 9th International Workshop on Ontology Matching Collocated with the 13th International Semantic Web Conference (ISWC 2014)*. 19

Euzenat, J., Shvaiko, P. *et al.* (2007). *Ontology matching*, vol. 18. Springer. 2, 13, 14, 15

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. 17, 27

Faria, D., Pesquita, C., Santos, E., Cruz, I.F. & Couto, F.M. (2014a). AgreementMakerLight: A scalable automated ontology matching system. *DILS 2014*, 29. 26, 40

Faria, D., Pesquita, C., Santos, E., Cruz, I.F. & Couto, F.M. (2014b). Automatic background knowledge selection for matching biomedical ontologies. *PLoS ONE*, **9**. 60

Gkoutos, G.V., Schofield, P.N. & Hoehndorf, R. (2012). The neurobehavior ontology: an ontology for annotation and integration of behavior and behavioral phenotypes. *Int Rev Neurobiol*, **103**, 69–87. 24

Gruber, T.R. (1993). A translation approach to portable ontology specifications. *Knowledge acquisition*, **5**, 199–220. 7

Guarino, N. (1998). *Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy*, vol. 46. IOS press. 7

Haendel, M.A., Gkoutos, G.G., Lewis, S.E. & Mungall, C. (2009). Uberon: towards a comprehensive multi-species anatomy ontology. In *International Conference on Biomedical Ontology (ICBO)*, Nature Publishing Group. 24

Hu, W., Chen, J., Zhang, H. & Qu, Y. (2012). Learning complex mappings between ontologies. In *The Semantic Web*, 350–357, Springer. 20

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. 17

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. 8, 24

Mungall, C.J. (2004). Obol: integrating language and meaning in bio-ontologies. *Comparative and functional genomics*, **5**, 509–520. 12

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. 12, 24

REFERENCES

Mungall, C.J., Bada, M., Berardini, T.Z., Deegan, J., Ireland, A., Harris, M.A., Hill, D.P. & Lomax, J. (2011). Cross-product extensions of the Gene Ontology. *Journal of Biomedical Informatics*, **44**, 80 – 86, ontologies for Clinical and Translational Research. 12, 44

Musen, M.A., Middleton, B. & Greenes, R.A. (2014). Clinical decision-support systems. In *Biomedical informatics*, 643–674, Springer. 9

Ngo, D. & Bellahsene, Z. (2012). Yam++: A multi-strategy based approach for ontology matching task. In *Knowledge Engineering and Knowledge Management*, 421–425, Springer. 17

Oliveira, D. & Pesquita, C. (2015a). Compound matching of biomedical ontologies. In *International Conference on Biomedical Ontology (ICBO)*. 4

Oliveira, D. & Pesquita, C. (2015b). Inferring logical definitions using compound ontology matching. In *International Conference on Biomedical Ontology (ICBO)*. 4

Otero-Cerdeira, L., Rodríguez-Martínez, F.J. & Gómez-Rodríguez, A. (2015). Ontology matching: A literature review. *Expert Systems with Applications*, **42**, 949–971. 16

Pesquita, C., Faria, D., Stroe, C., Santos, E., Cruz, I.F. & Couto, F.M. (2013). What's in a 'nym'? synonyms in biomedical ontology matching. In *The Semantic Web–ISWC 2013*, 526–541, Springer. 42

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*. 13, 19, 25, 31, 44

Porter, Martin F (2001). Snowball: A language for stemming algorithms. *https://snowball.tartarus.org/texts/introduction.html (visited: September 2015)*. 41

Ritze, D., Meilicke, C., Sváb-Zamazal, O. & Stuckenschmidt, H. (2009). A pattern-based ontology matching approach for detecting complex correspondences. In *ISWC Workshop on Ontology Matching, Chantilly (VA US)*, 25–36, Citeseer. 20

Ritze, D., Völker, J., Meilicke, C. & Šváb-Zamazal, O. (2010). Linguistic analysis for complex ontology matching. *Ontology Matching*, **1**. 20

Robinson, P.N. & Bauer, S. (2011). *Introduction to bio-ontologies*. CRC Press. 9, 11

Rosse, C. & Mejino, J.L. (2003). A reference ontology for biomedical informatics: the Foundational Model of Anatomy. *Journal of biomedical informatics*, **36**, 478–500. 24

Rubin, D.L., Shah, N.H. & Noy, N.F. (2008). Biomedical ontologies: a functional perspective. *Briefings in bioinformatics*, **9**, 75–90. 8

Schindelman, G., Fernandes, J.S., Bastiani, C.A., Yook, K. & Sternberg, P.W. (2011). Worm Phenotype Ontology: integrating phenotype data within and beyond the c. elegans community. *BMC bioinformatics*, **12**, 32. 24

Shrestha, R., Arnaud, E., Mauleon, R., Senger, M., Davenport, G.F., Hancock, D., Morrison, N., Bruskiewich, R. & McLaren, G. (2010). Multifunctional crop trait ontology for breeders' data: field book, annotation, data discovery and semantic enrichment of the literature. *AoB plants*, **2010**, plq008. 65

Shrestha, R., Matteis, L., Skofic, M., Portugal, A., McLaren, G., Hyman, G. & Arnaud, E. (2012). Bridging the phenotypic and genetic data useful for integrated breeding through a data annotation using the crop ontology developed by the crop communities of practice. *Frontiers in physiology*, **3**. 66

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. 11, 44

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. 24

Xu, L. & Embley, D.W. (2003). Using domain ontologies to discover direct and indirect matches for schema elements. In *Proceedings of the Semantic Integration workshop at the International Semantic Web Conference (ISWC)*, vol. 82, 97–102. 19