Domain-Aware Ontology Matching

Francisco José Quesada Real

Doctor of Philosophy

Artificial Intelligence and its Applications Institute
School of Informatics
University of Edinburgh

2020

Abstract

During the last years, technological advances have created new ways of communication, which have motivated governments, companies and institutions to digitalise the data they have in order to make it accessible and transferable to other people. Despite the millions of digital resources that are currently available, their diversity and heterogeneous knowledge representation make complex the process of exchanging information automatically. Nowadays, the way of tackling this heterogeneity is by applying ontology matching techniques with the aim of finding correspondences between the elements represented in different resources. These approaches work well in some cases, but in scenarios when there are resources from many different areas of expertise (e.g. emergency response) or when the knowledge represented is very specialised (e.g. medical domain), their performance drops because matchers cannot find correspondences or find incorrect ones.

In our research, we have focused on tackling these problems by allowing matchers to take advantage of domain-knowledge. Firstly, we present an innovative perspective for dealing with domain-knowledge by considering three different dimensions (specificity - degree of specialisation -, linguistic structure - the role of lexicon and grammar -, and type of knowledge resource - regarding generation methodologies). Secondly, domain-resources are classified according to the combination of these three dimensions. Finally, there are proposed several approaches that exploit each dimension of domain-knowledge for enhancing matchers' performance. The proposals have been evaluated by matching two of the most used classifications of diseases (ICD-10 and DSM-5), and the results show that matchers considerably improve their performance in terms of f-measure.

The research detailed in this thesis can be used as a starting point to delve into the area of domain-knowledge matching. For this reason, we have also included several research lines that can be followed in the future to enhance the proposed approaches.

Lay Summary

Despite the millions of digital resources that are currently available, their diversity and heterogeneous knowledge representation make complex the process of exchanging information automatically. Nowadays, companies are applying ontology matching techniques which try to find correspondences between the elements within different resources. Currently, these approaches are not enough in scenarios in which there are resources from different domains (e.g. emergency response) or when the knowledge represented is very specialised (e.g. medical domain), because matchers cannot find correspondences or find incorrect ones.

In our research, we have develop several solutions that allow matchers to take advantage of domain-knowledge to improve their performance. We have evaluated these solutions using two classifications of diseases, and the results confirm that matchers improve their performance using domain-knowledge.

Acknowledgements

I am extremely grateful, glad and proud of having this team of supervisors, who have guided and accompanied me during the whole PhD. I would like to thank them for their unwavering support, encouragement, tireless rereading of drafts, and their useful and thoughtful comments. Thanks to Alan Bundy, not only for his dedication and experienced advice, but also for being an example to follow even in the worse moments. To Fiona McNeill for her continuous help and being an inspiration as well as for introducing me into the ISCRAM community. To Gábor Bella, for his enriching discussions which always result in fresh and interesting ideas and for contributing by leading the implementation of Diversicon, which has become an indispensable tool in our research.

I would like to thank my examiners, Dr. Pavel Shvaiko and Dr. Nadin Kokciyan, whose insightful comments have improved the quality of the thesis.

This PhD was partially funded by the European Commission with the grant agreement No. 607062 (ESSENCE Marie Curie ITN). Thanks to Michael Rovatsos for making it possible and for his valuable feedback, before, during and after I got the position to start my studies at the University of Edinburgh. There are many people in ESSENCE who have helped me during my studies, I would like to specially thank Mila for making all the logistics easier.

To Fausto Giunchiglia for hosting me in Trento and making me feel as if I was at home, giving me the opportunity to present my research to the KnowDive members.

To Austin Tate, Ewan Klein and Michael Rovatsos for being on the panel of the different year reviews that I have had, providing enriching feedback.

I also wish to thank the members of the Resilience Department of the Scottish Government and physicians of the Andalusian Health Service for their interactions that have been vital in understanding such different domains.

Thank to all the folks of CISA and the DReaM group, where I have had the opportunity to learn and present my research in a friendly and relaxed environment. Special mention to Xue Li, for her feedback on the thesis and for being my liason with the university since I returned to Spain.

I would like to thank all who have contributed in my academic career, specially to Luis Martínez, who persuaded me to apply for the position at ESSENCE, and introduced me to the world of research.

Last but not least, I would like to thank my family for their constant unconditional support. It has been a difficult path and even now there are family members that are not among us, I am sure that they are sharing this joy with us.

Thank you very much.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Francisco José Quesada Real*)

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The need of communicating and sharing information is daily present in many different situations, for example with our family, friends or at work. Nonetheless, even though there are lots of contexts in which these activities are important, in those cases in which people's live are at risk they become critical, e.g. in Emergency Response (ER) or in health-related scenarios.

Technological advances have brought a wide range of possibilities and new scenarios, making easier the sharing of information with people who are located in different places. For example, executives of a multinational company can meet by video-conferences and work at the same time with the same documents no matter where they are. Thus, in order to take advantage of this technological revolution, most companies and governments are fostering the digitalisation of information with the aim of making it accessible to everyone just with a click. As a result of this process, in the last decades the number of digital resources has exponentially increased, and the data generated every day is more than 2.5 quintillion bytes [99].

At this point, we may think that the more available resources are, the better communication and data sharing we will experience. However, this is not totally true because even though we have access to millions of resources, each one is usually constructed for a particular purpose, representing knowledge attending to the particularities of that purpose. Therefore, after our research we claim that most resources represent knowledge differently. This fact makes difficult the automatic sharing of information. For example, if two regions of the same country want to merge their medical records, and their systems represent data differently, it is not possible to carry out this process automatically. As a result, the merging process will entail a lot of time and high labour costs trying to align manually the representations of these systems.

In order to solve these heterogeneity problems more efficiently, it is necessary to apply Ontology Matching (OM) techniques, with the aim of aligning the entries of the different resources.

1.1 Motivation

Although the application of OM techniques solves many heterogeneity problems, there are still cases in which the current approaches present some limitations, not being able to identify correspondences between resources or identifying some of them incorrectly. Examples of these cases can be found in ER scenarios and in the medical sector.

In the former, it is not unusual the collaboration of response agencies from different areas of expertise, with the aim of giving the best response and restoring the situation as soon as possible. This means that participants are likely to share information and digital resources which represent different knowledge. For example, the knowledge of the roads of a region represented by the fire brigade and the ambulance service will be different to the one represented by the police. Nonetheless, despite this heterogeneity, the exchange of information between these agencies is vital, for example during an evacuation. In this scenario, the police can provide the fire brigade with information of the roads that are blocked, allowing them to choose alternative routes that avoid the block and so, optimising the evacuation. In some cases the process is even more complex because ER scenarios some times imply that the participant agencies have not worked together before, so we have to assume that their digital resources are not pre-aligned. Therefore, it is necessary to apply approaches which allow agencies to align their resources quickly and with high degrees of precision and recall.

In the latter, all resources belong to the same domain, so we might think that the complexity of aligning resources might be lower than in ER scenarios where resources are from different areas of expertise. This is true at certain degree because within the same domain it is more likely that resources share some representations of knowledge. However, there are many cases in which the way in which knowledge is represented differs from one resource to another. Some causes of this diversity are:

- *Medical specialisations.* The medical domain is an extremely broad area of knowledge that is divided into different specialisations with the aim of delving into each sub-area. Thus, the knowledge represented in a *cardiology* resource will be different than the one represented in an *orthopaedic medicine* resource.

- *Standards.* Currently there are different classifications of diseases that are used by medical practitioners depending on their region or medical centre. This makes that two hospitals of the same city might use different descriptions to catalogue the same disease.

- *Locality factors.* It is common that medical centres use a standard classification of diseases, as described above, and in some cases, their particular terminology to represent the knowledge that they commonly use. The aim of defining these new terms is to facilitate the work of their medical practitioners, so it is not expected the terminology to be used outside its application hospital. For example, the glossary defined by the Mayo Clinic [32], is applicable to all the medical centres of this institution, but not to other hospitals.

The described scenarios add complexity to the matching problem, being extremely difficult to completely align diverse resources. The main problems can be divided into two categories:

- *False negatives.* The matcher does not have enough information to conclude that there is an alignment between entries of different resources. Therefore, it is not possible that the matcher can output the correspondences between ER resources that represent the command & control levels (*gold, silver, bronze*, or *strategic, tactical* and *operational*) because the needed knowledge is not in matcher's Knowledge Base (KB).

- *False positives.* The matcher outputs wrong mappings between the input resources. Examples of these cases might be ambiguity problems in which two terms are mapped because they are homonyms, no matter the domain to which the resources come from and so, the sense of each term. For example, the term "*cruse*" has two senses:

    1. *Small jar to hold liquids (oil or water).*

2. *Charitable organisation offering bereavement counselling, advice and support throughout the United Kingdom (UK).*

   As we can see, in most cases when two resources have the term "*cruse*", they will match correctly as they refer to the jar. However, if the resource belongs to any ER agency of the UK, it is likely that "*cruse*" refers to the second sense.

In the literature we can find multiple approaches which use matchers' Background Knowledge (BK) to carry out the OM process. Indeed, depending on the scenario in which the matcher is applied, the BK represents either general-purpose knowledge (e.g. WordNet (WN)) or knowledge specific to the particular domain of the scenario (e.g. architecture, geography...). Moreover, all of these approaches only focus on the representation of symbolic knowledge to perform the matching process, not considering other issues such as domain grammar.

Knowing the above problems and the limitations of the current approaches, we focus our research on studying how domain-knowledge can be used by matchers to improve their performance.

1.2 Research Aims

Our hypothesis is that:

When matching domain ontologies, matchers with Domain-Aware (DA) functionality, have a better performance in terms of precision and recall than those which do not have this functionality because domain knowledge helps them to disambiguate and discover mappings that otherwise could not be found, and reject mismatches that look superficially plausible.

The key aims of our research are:

1. To provide a taxonomy of kinds of domain-knowledge, attending to different dimensions.

2. To provide guidelines to formalise domain-knowledge depending on its nature.

3. To add DA functionality to matchers by integrating domain-knowledge into matchers.

4. To evaluate which kind of domain-knowledge is most beneficial for each kind of matching process.

All these aims have been achieved, and are discussed in detail in chapter 7.

1.3 Evaluation Results

Following the motivational examples, we tried to evaluate our approach both in the ER and the medical domain. It is necessary to highlight that the former is a domain in which it is difficult to get data for testing. The main reason is due to privacy issues, because apart from using sensitive data, these organisations normally have internal policies that do not allow sharing their resources with people outside their consortium. In our case, we experienced these difficulties and despite the willingness of the practitioners from the resilience department of the Scottish Government, and the 112 and 061 Jaén's coordination centres (Spain), we could not obtain access to ER resources for testing our experiments. For this reason, the evaluation contained in this thesis is applied only to the medical domain.

Anyway, this is a suitable choice because this domain contains clear examples of heterogeneity problems that professionals daily have to deal with and it also provides a gold standard to evaluate against. In particular, the evaluation of our hypothesis is carried out by an experiment which lies in matching a subset of two of the most extended classification of diseases: The US Diagnostic and Statistical Manual of Mental Disorders (DSM-5) and International Classification of Diseases, version 10 (ICD-10).

The matching process is carried out four times for each matcher. Firstly, it is executed using the matcher's vanilla version whose KB only has general knowledge (concretely we use WN). Secondly, the matcher carries out the matching process having a KB that includes the vanilla's knowledge plus a symbolic medical extension. Thirdly, the KB of the vanilla version is enriched with a medical grammar extension, and finally, for the last execution the matcher includes in its KB all the extensions.

The performance of each matching process is evaluated attending to the most used metrics for Information Retrieval (IR) evaluation: *precision*, *recall* and *f-measure* [11]. In order to compute these metrics, we use as our gold standard the mappings between both classifications that are generated and released by the American Psychological Association (APA). In principle, we have followed the APA gold standard strictly, but after analysing the results in depth we have discovered that our results were penalised for finding correct mappings that are not included in the gold standard. After checking these findings with different health professionals we decided to extend the gold standard with these correct mappings.

In our research, we carried out the experiment with two different matchers: the Semantic Matcher (S-Match) [58] and the Logic-based and Scalable Ontology Matching (LogMap) [79]. These matchers are two of the most relevant ones in the OM community as we can see in the hundreds of cites that their papers have. The main reason for using these two matchers is because they perform the matching process in a different way, so analysing both methods allows us to identify which aspects affect each kind of matching and abstract the general features (for an extended comparison of these matchers see Section 2.3). Thus, these features can be generalised for the rest of the matchers. The main difference between these matchers is that whereas S-Match carries out semantic matching to find semantic relations between the elements of the input ontologies, LogMap uses context similarity measures to find them.

The results of the experiments suggest that our hypothesis is true as S-Match and LogMap improve their *f-measure* when using domain-knowledge.

The application to other domains is feasible as the medical domain entails all the complexity that can be found when representing domain-knowledge. However, the costs in terms of time and effort will depend on the kind of resources that need to be used. For example, the generation of a symbolic extension following a fine integration process (see Section 4.4.1.2) is harder than a rough integration (see Section 4.4.1.1), or the integration of statistical resources. On the other hand, every resource can be integrated via fine integration, but for a rough integration it is necessary to be represented as a taxonomy, and for the generation of statistical resources it is necessary to exploit huge amounts of data which in some cases do not exist.

1.4 Organisation of Thesis

The thesis is organised as follows:

Chapter 2 contains the literature survey. In this chapter, our research is contextualised, going from Knowledge Representation (KR) to the need of applying OM, where we identify the research gap that motivates our work to add DA functionality to matchers. Finally, we describe the characteristics of the domains in which our work has been applied. These domains show the complexity of the matching process in ER scenarios, that involve participants from different areas, and within the medical domain, where knowledge is usually represented with different degrees of specialisation. Thus, the methodology applied in these domains, can be extended to other domains.

Chapter 3 details some basic concepts of OM that are necessary to understand our approach. Besides, the matchers that have been used are described, as well as other third-party resources: integration tools, domain-knowledge resources and the classifications of diseases used for the evaluation.

Chapter 4 is the core of the thesis. In this chapter, the research contribution is explained in detail. Firstly, it is provided a taxonomy of domain-knowledge according to three dimensions: *specificity* - degree of knowledge specialisation -, *linguistic structure* - role of lexicon and grammar - and *types of knowledge resources* - regarding generation methodologies-. After that, domain-knowledge resources are organised attending to the combination of the three dimensions. Finally, there are provided several guidelines for taking advantage of domain-knowledge dimensions in OM.

Chapter 5 details the implementation carried out, highlighting the domain-knowledge extensions that have been produced or adapted from existing resources.

Chapter 6 evaluates the hypothesis, analysing in depth the results obtained after running the evaluation experiments. From the analysis we can see how DA knowledge contributes to improving the performance of matchers in terms of precision and recall. Moreover, enriching matchers' BK with domain-knowledge benefits more to those matchers which not only use string similarity measures to find mappings, but also those which take advantage of semantic relations.

Chapter 7 summarises the thesis highlighting the conclusions and the future works. The main conclusion of the thesis is that DA matching has a better performance than the traditional matching processes in terms of *f-measure*, mainly because we are considering a more developed concept of domain-knowledge, not only limiting it to domain ontologies. Therefore, matchers can take advantage of the three dimensions of knowledge that we have identified (specificity, linguistic structure and types of knowledge resources). Nonetheless, there are also some limitations that produce that *precision* and *recall* do not increase at the same time, so when one increases, the other slightly decreases. As future works there are different research lines to improve the current results, focusing on addressing the detected limitations. For any of these options, this thesis will be helpful to other researchers as a starting point.

1.5 Summary

In this introductory chapter, we firstly explained the motivation of our research, establishing the hypothesis that this thesis explores. After that, the evaluation plans were outlined. The chapter finishes with an outline of the thesis, describing briefly each chapter.

Chapter 2

Literature survey

In this chapter, we analyse the state of the art of the OM field, and identify the gap that motivates our research. First of all, there are introduced some basic concepts of knowledge representation, focussing on ontologies and the challenges of sharing data between diverse ontologies. After that, we present OM as the solution to address these challenges. Despite having minimised many of the challenges, there are other issues which still need to be tackled, and which inspire the research carried out in this thesis. The chapter finishes with a description of the ER and the medical domains, which we have used to evaluate our research.

2.1 Knowledge Representation

The research in formal Knowledge Representation (KR) began hundreds of years prior to the appearance of the first computer [26]. Nonetheless, it was not until the end of the 1960s when the interest in this area drastically increased, motivated by early discussions of representing knowledge in Artificial Intelligence (AI) [8]. The need of representing knowledge has been present since then until our days, being a key task for designing intelligent systems [24].

Among the different structures used to represent knowledge, ontologies have acquired a considerable relevance, mainly because they focus on knowledge sharing [126, 139]. In the literature, we can find many definitions of *ontology*, one of the most accepted being proposed by Gruber who defines an ontology as "*a formal, explicit specification of a shared contextualisation*" [64]. Another relevant definition was proposed by Zhao et al., who consider ontologies as "*models of aspects of reality which define vocabulary, concepts, relations and meanings for a* *specific domain*" [152]. Although the origin of the term in the Ancient Greece, refers to *ontology* as a "*general purpose classification or taxonomy of knowledge*" [123], from the previous definitions we can infer that KR researchers are not aspiring to produce such a general ontology because it is impossible to generate one ontology which includes all the knowledge currently known.

Nicola Guarino, identifies different ontologies according to their level of expressivity [66, 67] (see figure 2.1):

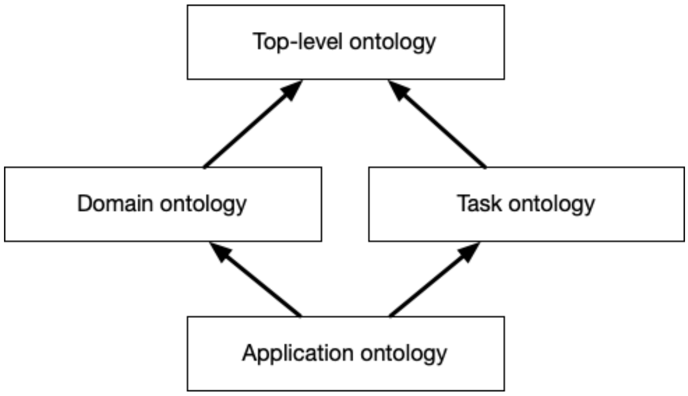

Figure 2.1: Kinds of ontologies according to their level of expressivity. Extracted from [67].

- *Top-level ontologies (upper ontologies).* Describe general concepts that are independent of a problem or a domain. Therefore, a same top-level ontology that represents concepts such as event, action, space, time or object, might be used as a basis for representing other ontologies that are more specific [106].
- *Domain ontologies* and *task ontologies.* Specialise a top-level ontology with vocabulary of a general area of knowledge, such as medicine [107] or cultural heritage [71], or a general task or activity such as negotiating or selling [140].
- *Application ontologies.* Describe concepts that specialise both domain and task ontologies. These concepts usually correspond to roles that users have to perform when carrying out a particular task. An example, are the ontologies proposed by Mata et al. to allow intelligent agents to communicate and carry out a consensus reaching process [100].

The design and development of ontologies are not trivial tasks, requiring a deep understanding of the knowledge to be formalised and a profound thought of which structure is most suitable for representing that knowledge. There are many researchers who have contributed with guidelines and recommendations to carry out this process [65, 67, 68, 69, 132, 135], which is denoted as *ontological engineering* [45].

Thomas Gruber identifies the following preliminary set of design criteria for creating ontologies [65]:

- *Clarity.* An ontology should convey the intended meaning of the represented terms. Definitions should be objective, formalised by logical axioms and documented with natural language. Complete definitions (predicate defined by necessary and sufficient conditions) are preferred over partial ones (include either only necessary or sufficient conditions).

- *Coherence.* The axioms defined should be logically consistent, so that they allow inferences consistent with the definitions.

- *Extendibility.* The ontology should be designed to allow the definition of new specialised terms, extending it monotonically. That is, not requiring the revision of the existing definitions when new terms are added.

- *Minimal encoding bias.* The conceptualisation should be specified at the knowledge level, not depending on a particular encoding language.

- *Minimal ontological commitment.* An ontology should include the sufficient knowledge to allow communication between the agents that use it. A way of minimizing ontological commitment is by defining the essential terms that allow the communication knowledge with the weakest theory.

Ontology design requires making trade-offs between the above criteria in order to adapt it to our needs. For example, we could weight extendibility more than minimal ontological commitment if the former is more important than the latter for the scenario in which the ontology will be used.

Similarly, Nicola Guarino proposes several design principles. In this case, with the aim of solving *is-a* overloading problems [68]:

- *Be clear about the domain.* It is necessary to clearly identify in advance the entities that take place in a domain, with the aim of formalising the theories about such domain.

- *Take identity seriously.* Based on the Lowe's principle ("*No individual can instantiate both of two sorts if they have different criteria of identity associated with them*" [91]), identity criteria play an essential role in identifying ontological distinctions.

- *Isolate a basic taxonomic structure.* An ontology should have a basic backbone of categories and types that form a tree of mutually disjoint classes.

- *Identify roles explicitly.* Tagging roles explicitly allows to easily isolate the basic backbone, and to carry out inferences that involve mutual disjointness while avoiding explicit declarations.

Despite the above mentioned recommendations, in the ontological engineering process, the subjectivity of the participant engineers plays a vital role, so we have to assume that different teams of engineers formalising the same knowledge may produce subtly different ontologies. Chocron et al. propose a way of constructing ontologies applying crowdsourcing approaches. Specifically, they take advantage of the data obtained from the users when they interact with a gamification system. After that, an ontology engineering process is carried out [29].

Carrying out an ontology engineering process it is possible to construct diverse ontologies. Uschold and Gruninger identify different kinds of ontologies according to their degree of expressivity [139] differentiating between: glossaries, data dictionaries, thesauri, taxonomies, metadata, data models and formal ontologies.

The diversity of ontologies and the heterogeneity of KRs add complexity to the task of sharing data. Therefore, uniformity of terms between ontologies is not possible, as shown in [126]. In fact, this uniformity might be only useful in a specific domain, but really challenging for the development of a "neutral and common framework for all descriptions" [131]. In order to allow data sharing between ontologies, researchers have proposed different techniques to align ontologies. The field in which experts do research on how aligning ontologies with different KR is called OM.

2.2 Ontology Languages

Ontology languages allow users to write explicit, formal conceptualizations of domains models [97]. Below there are detailed some of the most relevant ontology languages for our research [35].

2.2.1 OWL

The Web Ontology Language (OWL) [13, 102] is used to describe ontologies over the web, being currently one of the most popular standard to do so. It is built on the Resource Description Framework (RDF) [96] and Resource Description Framework Schema (RDFS) [6]. Moreover, it provides a vocabulary for describing properties and classes.

In RDF each piece of information is represented as a triple composed by a property connecting two resources: (i) the *subject*, and (ii) the *object* (i.e: *subject:* Harry - *property: isBrotherOf* - *object*: William)

RDFS (methodological knowledge) provides basic constructs to define an ontology (conceptual knowledge) in order to specify RDF real data (factual knowledge); in particular it allows to define classes, properties, and their subsumption hierarchies along with the domain and the range of each property. OWL was born from the need to extend RDFS to increase its expressivity.

OWL languages are based on description logics. Three different OWL sublanguages have been defined, with a growing degree of expressivity (see Figure 2.2):

- *OWL Lite* provides only simple constructs to describe domains. Therefore, its expressivity is limited. The original intention was to allow users to construct basic hierarchical classifications with simple constraints (e.g. cardinality constraints).

- *OWL DL* aims at providing the maximum expressiveness while preserving computational completeness, decidability and reasoning capabilities. These characteristics makes this sublanguage the most popular and currently used. Despite it includes all *OWL DL* language constructs they can only be used under certain restrictions.

- *OWL Full* adds further expressivity to *OWL DL*. It is very expressive, but it is not decidable, so although it can support automatic reasoning it is not possible to place time limit on it without losing completeness. It is also compatible with RDF Schema.

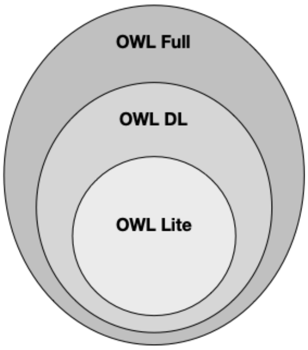

Figure 2.2: OWL sublanguages (extracted from [97]).

The OWL Lite format is used in our research for the evaluation with LogMap because this matcher needs the input datasets in this format. To do so, we have transformed into OWL Lite the medical datasets that we use.

2.2.2 XML Schemas

XML Schemas [38] have been introduced for specifying the structure of Extensible Markup Language (XML) documents [25, 34]. The main components of XML schemas are elements, attributes, and types. Elements can be either complex for specifying nested subelements, or simple for specifying built-in data types, such as *string* for an element or attributes. Figure 2.3 shows an example of a XML schema of the ER domain.

Even if element definitions can be extended or restricted as subcategories of a classification, the emphasis is on the structure: the extension of an element is made by providing the elements which are modified in this structure.

It is necessary to consider that XML schemas are defined according to which future documents will be created, as oppose to an ontology.

The specialisation hierarchy in XML schema defines which kind of elements can occupy the place of another kind. For instance, if the element "*hospital*" contains, "*doctor*" then adding "*radiation oncologist*" to this "*hospital*" is authorised. In principle, this classification structure does not have to correspond to any natural classification of the objects.

In our research, the datasets that we use are translated as XML schemas, which is S-Match input format .

```xml
<?xml version="1.0" encoding="UTF-8"?>
<context>
  <node id="n0_7502794">
    <name>Top</name>
    <children>
      <node id="n1_7502800" parent-id="n0_7502794">
        <name>emergency</name>
        <children>
          <node id="n10_7502875" parent-id="n1_7502800">
            <name>significant emergency</name>
          </node>
          <node id="n9_7502855" parent-id="n1_7502800">
            <name>catastrophic emergency</name>
          </node>
          <node id="n8_7502850" parent-id="n1_7502800">
            <name>local emergency</name>
          </node>
          <node id="n7_7502840" parent-id="n1_7502800">
            <name>serious emergency</name>
          </node>
        </children>
      </node>
      <node id="n20_7502977" parent-id="n0_7502794">
        <name>command and control level</name>
        <children>
          <node id="n23_7503002" parent-id="n20_7502977">
            <name>tactical</name>
          </node>
          <node id="n22_7502996" parent-id="n20_7502977">
            <name>operational</name>
          </node>
          <node id="n21_7502984" parent-id="n20_7502977">
            <name>strategic</name>
          </node>
        </children>
      </node>
    </children>
  </node>
</context>
```

Figure 2.3: Example of a XML Schema of the ER domain.

2.2.3 LMF

The Lexical Markup Framework (LMF) is a model that provides a common standardized framework for the construction of Natural Language Processing (NLP) lexicons [47, 48]. The main goals of LMF are to provide a common model for the creation and use of lexical resources, to manage the exchange of data between and among these resources, and to enable the merging of large number of individual computational resources to form extensive global digital resources.

Different LMF instantiations can include monolingual, bilingual or multilingual lexical resources. Moreover, they cover all natural languages, not being restricted only to European languages. To do so, they use Unicode to represent all kinds of scripts and orthographies.

Regarding the structure of a LMF file, it follows the XML format. In particular, it defines a general element called "*Lexicon*" which is the container of all "*lexical entries*" in a language. The "*lexical entry*" is a container for managing the top level language components. Therefore, there will be as many "*lexical* *entries*" as number of different words, expressions and affixes in the lexicon.

In our research, we use LMF to develop the extensions that contain the knowledge from the ER and medical domains.

2.3 Ontology Matching

Figure 2.4 depicts two extracts of the glossaries used by two ER agencies: *ER Agency A* and *ER Agency B*. As we can see, it is complex to share data automatically between these organisation because, for example, either they have a different number of emergencies, or they represent them using distinct and non-synonymous terms. Thus, it is necessary to carry out an OM process that generates a set of correspondences between the entities of the different ontologies [38].

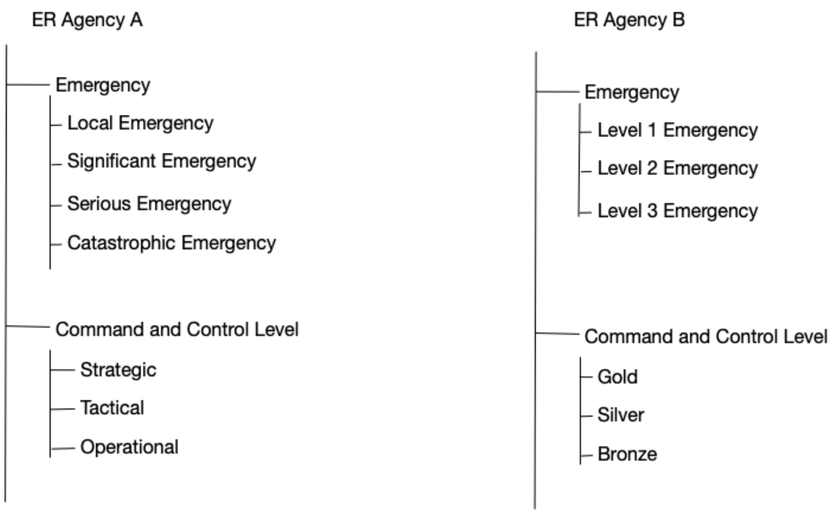

Figure 2.4: Example of two glossaries of ER agencies.

As a result of the OM process, we can see how "*Emergency*" and "*Command and Control level*" (ER Agency A) has a correspondence with "*Emergency*" and "*Command and Control level*" (ER Agency B). However, the other entries do not have correspondences, or they are not clear enough. These are the kind of questions that researchers in the OM field try to answer.

In the literature, we can find a wide range of methods and approaches to carry out OM [49, 117, 136].

The variety of methods and the number of multiple ontologies make it difficult to choose the most suitable alignment between them. Nefzi et al. address this problem using multi-criteria methods to select the best mappings between different ontologies [111].

There are approaches which carry out a negotiation process between intelligent agents to generate the output of the matching process, for example identifying potential violations within ontologies and automatically repairing them by negotiating [81]. Similarly, Chocron et al. use agents for evaluation purposes as an alternative to the traditional evaluation measures, determining the correctness of the alignments according to what extent agents understand each other with the mappings in each interaction [30].

Among all the proposals and the approaches, it is widely believed that the use of Background Knowledge (BK) is necessary to discover the mappings between the input ontologies [76]. Indeed, it is proved that using domain ontologies, such as BK, helps to overcome matching problems [3, 124]. Following this idea, Annane et al. present a two-step process to select and build a BK from parts of external ontologies, only including concepts related to the ones to be matched. This process focuses on improving efficiency, by matching only subsets of concepts of the external ontologies, instead of matching the complete resources, but not losing effectiveness in terms of precision and recall [9].

Hetch et al. propose an approach that refines alignments obtained from current OM methods, by taking advantage of the links between ontologies represented in Linked Open Data (LOD).

In recent years, the increase of large-sized knowledge resources has entailed the application of OM processes in order to integrate these resources. However, OM processes have experienced different challenges such as the increase of complexity and execution time, or the need of more memory. In order to address these issues, researchers are applying different techniques to OM tools to achieve scalability. Some of these techniques are the reduction of search space, parallel composition and multiple matcher combination [114].

In general, OM technology corresponds to finding an isomorphism between the sub-graphs [147], however the process of matching two ontologies is a complex and time consuming task, because it is a non-linear problem and it grows exponentially according to the number of entities to match. For this reason, some researchers are using evolutionary algorithms as an efficient approach to tackle these problems [146]. Mainly, their approaches focus on either optimising the parameters involved in the OM process [1, 142, 148], or optimising the set of correspondences that is output by the matcher [2, 128, 143].

In contrast to the previous approaches, which perform the matching process automatically, Lambrix and Kaliyaperumal propose a framework that focusses on involving users as a key part of the alignment process. User involvement is a challenging task, so that, the framework offers three kind of sessions that can be interrupted and resumed at any time, in addition to different ways of user assistance during the alignment process. Thus, the framework allows them to suggest, recommend and validate mappings [85]. In this approach, Li et al. highlight that the profile of the user, the services of the alignment system, and the system's user interface are three aspects which directly affect user validation [89].

As we have seen, since the beginnings of the OM field, there have been proposed multiple matchers, which apply different algorithms, methods and parameters to carry out the matching process.

The OM community runs every year the "Ontology Alignment Evaluation Initiative (OAEI)" [1] to forge consensus and improve the quality in the evaluation of matchers. In addition to achieving this goal, the event is the best scenario for researchers for presenting their matchers. Thus, any matcher can participate in the yearly OAEI in order to compare its performance, when aligning the proposed datasets, with respect to other matchers. Nonetheless, it is not mandatory that any new matcher must to participate in the OAEI, so there are cases of matchers that, even though they implement interesting matching algorithms, do not participate in this event. An example of these matchers is S-Match, which currently is not adapted to handle large datasets such as the ones used in the OAEI.

Below, there are briefly described three of the most relevant matchers according to their impact on the OM community, regarding number of citations and performance:

- *S-Match* [58]. S-Match explores two input graph structures exploiting the information codified in the nodes and the structure of the graph, outputting, as a result, the logic relations between the nodes of the structures. The matching process involves the transformation of each node into logical formulas and finding the concepts within the formulas in the matcher's KB. Figure 2.5 shows an example of the output of S-Match after matching

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

two ontologies. We can see how the matcher recognises equal mappings (drawn in green), and the relations more general (drawn in blue) and less general[2]. An interesting point of this matcher is that it can deal with lack of knowledge by carrying out the matching process using semantic relations.

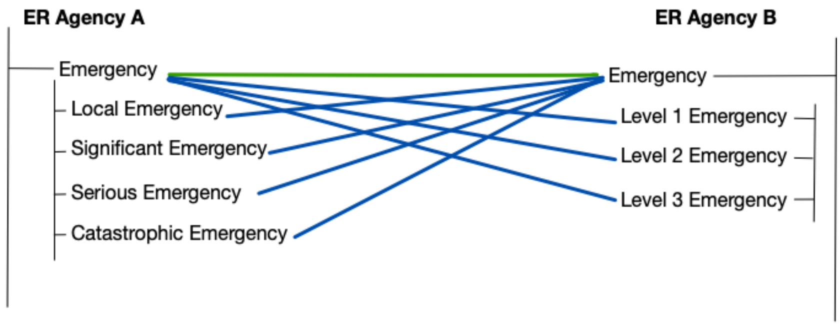

Figure 2.5: Example of semantic matching.

- *LogMap* [79]. LogMap is a matcher optimised for matching ontologies with hundreds or thousands of entities. Its matching process involves: the indexation of the input ontology labels, the computation of initial mappings, and a refinement process to maximise precision and recall. LogMap is one of the matchers which have the better performance in the OAEI [4, 5].

- *AML* [43]. The AgreementMakerLight Ontology Matching System (AML) is a framework optimised to match large ontologies efficiently. To do so, it incorporates in its OM module a variety of matching algorithms, selection algorithms, and a data structured based on MultiMaps to store mappings between the input ontologies. Currently, this is the matcher that performs the best on average in the OAEI [4, 5]. However, the combination of different matching algorithms, makes it more difficult to trace the impact of our approaches than in a matcher that operates with a single matching algorithm. For example, the output of one algorithm might be overwritten by the following algorithm. Considering that this requires a deep analysis and the limited time of the PhD, we have decided to postpone the integration of our approach in AML as future work.

[2] By default, S-Match identifies reciprocal semantic relationships, so not only "*Emergency*" (ER Agency A) is identified as *more general* than "*Level 1 Emergency*" (ER Agency B), but also "*Level 1 Emergency*" (ER Agency B) is identified as *less general* than "*Emergency*" (ER Agency A). For simplicity and to make the figure legible, there are represented only the minimal relationships between the elements of both ontologies.

The main reasons for choosing S-Match and LogMap stem from the particularities of each matcher that result in different ways to carry out the matching process. Regarding the former, it employs semantic matching, outputting at the end of the process all the semantic relations that the matcher has identified (equivalences, subsumptions, and disjoint). Apart from that, S-Match uses domain-independent knowledge in its BK. Regarding LogMap, it carries out the matching process applying reasoning and diagnosis tasks. In this case, all outputs have associated a confidence degree, which indicates to what extent the output is likely to be a mapping or not. If the confidence degree of an output is above a defined threshold, the mapping is considered, in other cases, it is discarded. Concerning the BK, LogMap uses some of the biomedical resources contained in the Unified Medical Language System (UMLS).

Both matchers S-Match and LogMap are described in detail in sections 3.1.2.1 and 3.1.2.2, respectively.

As has been discussed above, the current matchers solve many problems caused when the integration or data sharing of heterogeneous resources is necessary. However, several years ago, Shvaiko and Euzenat identified several challenges that remain to be addressed [129]:

1. *Large scale evaluation.* The evaluation of OM with large datasets implies that gold standards need to be generated automatically, because doing that manually entails a huge effort as the datasets become larger. Apart from that, it is interesting to define more accurate evaluation quality measures that allow us to assess how good is a matcher for a particular application, not only focusing on precision and recall.

2. *Efficiency.* This is a critical issue, especially when the user requires a quick response or when there is a memory limitation. So, it is necessary to address scalability in OM solutions.

3. *Matching with BK.* An alignment between two ontologies can be carried out by having a BK for the ontologies and extracting relations between ontology entities. Adding new knowledge is challenging because it may help to retrieve new information (increase recall), but these new information may be incorrect (decrease precision). Enriching matcher's BK is specially important in domain-specific scenarios, because the number of domain-specific terms is normally high. However, this fact also presents

a dilemma that involves the effort/cost of developing the extensions and integrating the new knowledge into the BK, and keeping the BK coherent.

Regarding the former, it is necessary to say that extending a BK with highly specialised terms, may imply a huge effort, and it is possible that it does not have a significant impact on the matching process. As for the latter, if the new terms entail duplicating lexical entries with slightly different meanings, it is likely that in the end the BK gets redundant causing small mismatches.

4. *User involvement.* In some cases, the results of the matching process need to be validated by humans, so the larger the size of data is, the more difficult this task will be. Therefore, it is necessary to propose ways in which users can be involved in the OM process without being overwhelmed by the amount of data. This challenge is even worse in stressful scenarios, such as in ER. In these cases, practitioners might only have limited time to spend in the OM tasks, so it is essential to provide the best user experience as possible. For example, it is necessary to improve the scalability of the visualisation in order to avoid that the user gets lost within an enormous amount of data.

5. *Explanations of OM.* Apart from providing the computed alignment, OM systems sometimes need to give explanations of the results. Thus, users will clearly understand the meaning of the alignments and use this information for decision making. Addressing this challenge is vital in ER scenarios where practitioners have to make decisions as quick and best as possible. For example, it might be useful to provide an interactive environment to help users accept or revise suggested correspondences.

6. *Collaborative and social OM.* Information obtained both, explicitly or implicitly by social interaction may help to improve alignments. However, it is necessary to deal with incomplete and inconsistent alignments after applying the obtained information, as well as malicious users. Other important aspect are the incentives that drive people to collaborate, because collaborative matching depends on the creation of a critical mass which support the matching task.

7. *Alignment infrastructure.* In order to store and share alignments it is necessary to provide convenient and interoperable support. This involves

the use of standards to communicate and retrieve alignments. However, one of the major problems is that currently matchers do not produce output files in the same format. For example, S-Match outputs a set of semantic relations whereas LogMap generates the tentative mappings with a confidence value. Addressing this challenge will help the exchange of information, specially in dynamic environments where ontologies evolve fast, because as soon as a new resource is aligned, these alignments can be shared and all partners may benefit from them.

We have seen some recent publications in which researchers propose methods to address some of these challenges, such as *user involvement* or *efficiency*. In our case, our hypothesis is partially focused on challenge number 3: "*Matching with BK*", in which authors highlight that enriching BK with new information might negatively affect precision, so this issue needs to be taken into account in our evaluation.

In addition to the previous challenge, we have also detected that there are some scenarios in which the use of BKs is not enough to solve problems such as ambiguity, specialisation or failing to find relevant mappings [124]. The common denominator of these scenarios is that they involve a huge amount of domain-knowledge, which is highly specific. Knowing, that we can infer that these problems might be for two causes:

1. *BK not specialised enough.* Despite having specific domain-knowledge, the BK still needs more detailed knowledge to have represented at least the same knowledge that is within the ontologies to be matched.

2. *Lack of domain independent knowledge.* The matching process requires general purpose knowledge that is not represented in the BK. An example might be a BK that exclusively represents knowledge from the Architecture domain and that is used to match two knowledge resources from the Architecture domain that include descriptions. In cases when descriptions only use terms from the Architecture domain the matcher will discover the mappings, but if the descriptions use general terms that are not included in the BK, the matcher will not be able to find these mappings.

An idea to address the above problems, might be the combination of both kinds of knowledge, domain-independent and domain-specific. Regarding the latter, it is also necessary to take into account the degree of specialisation of the domain-knowledge that will be required in the OM task, in order to avoid the problem of not having a BK specialised enough.

In the literature, we can find several proposals which attempt to solve these problems by using domain awareness, that is by identifying the domain of the ontologies to be match and using the knowledge of this domain that is represented in the BK [130]. This approach, solves some ambiguity problems, but when the knowledge is not included in the BK because it is too specialised, the heterogeneity problems produced by that specific knowledge cannot be addressed.

One example of these works was proposed by McCrae et al., who used domain adaptation to improve the performance of adapting an ontology into a different cultural context in multilingual problems. Thus, they construct a taxonomy of domains, associating different terms with several translations to each domain. Once they detect the domain of a sentence, they search for a translation in a specialised machine translation resource [101]. In a similar way, León-Araúz and Farber present an approach to describe concepts and terms of cross-lingual correspondences, considering domain and cultural constraints [86]. Giunchiglia et al. define a knowledge framework organised into a number of facets by defining one or more domains [53, 55]. The main purpose of this framework is to make easier the development of ontologies and facilitate matchers to use domain-knowledge. Thus, based on these ideas, Giunchiglia et al. developed a large-scale geo-spatial ontology [54] defining also, some guidelines for ontological construction.

In order to take advantage of new domain-knowledge, it is necessary to add it into the matcher's BK or have several BKs. However, as we mentioned before, it is interesting the idea of having a BK with both, general and domain-specific knowledge. One of the most popular domain-independent resources used as matcher Knowledge Base (KB) is WordNet (WN) [44]. In this lexical resource, nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms, called *synsets*. This way of grouping synonyms, is useful for matchers, which can discover that there is a mapping between two terms if both belong to the same synset. For example, if *Ontology A* has the entity "*house*" and *Ontology B* has the entity "*dwelling*" the matcher will output that there is a mapping between both entities as *house* and *dwelling* belong to the same synset.

There are some works focused on providing WN with domain information. Magnini and Cavaglià integrate subject field codes into WN, annotating noun synsets, which is useful for grouping synsets of the same domain [93]. Likewise, Bentivogli et al. develop the WN Domain Hierarchy, which is a language independent resource, composed of 164 domains such as Architecture, Sport or Medicine [18]. In the same way, Strapparava and Valitutti propose tagging synsets which represent affective meanings, in order to provide a lexical representation of affective knowledge. González et al. improve WN domains by an automatic graph-based method which propagates domain information through the knowledge base [60, 61]. Also, Gella et al. construct a domain-specific and multilingual corpora aligning WN domains and topics with Wikipedia categories [51]. There are many examples of domain-knowledge resources integrated in WN coming from different areas such as the maritime domain [122], the architecture and construction domain [17], the meteorology domain [98] or the bioenergy domain [42], among others.

The previous domain-knowledge resources have helped researchers to use WN domains to carry out Word Sense Disambiguation (WSD) [84, 141]. Namely, they follow the ideas of Gale et al. who claim in [50] that if a polysemous word appears many times in the same discourse, it is extremely likely that these words will all share the same sense. This idea is also extensible to collocations [149]. Thus, if the word "*bank*" appears in a sentence that belongs to the geographical domain, following these ideas is possible to disambiguate and say that "*bank*" refers to a sloping land, rather than a financial building.

Following these ideas, Bella et al. use domain-knowledge for WSD between multilingual resources in [16], finally extending the approach to multilingual OM by carrying out a two-step process that firstly processes multilingual natural language and then matches using language-independent and domain-aware background knowledge [14].

Despite the previous proposal and even though in many cases the use of domain-knowledge avoids some failures, there are still problems that need to be addressed. Mainly we focus on reducing the number of false negatives (undiscovered mappings) and false positives. These problems are caused when matching both, resources of the same domain, but with different degrees of specialisation (e.g. ontologies of a hospital and the ontology of a local surgery), or resources from different domains (e.g. ontologies of the police and the ambulance service). Moreover, so far, all the domain-knowledge approaches limited the BK enrichment to only include sets of entities belonging to a particular domain, but not considering other facets, such as the particularities of the grammar of a specific domain. These facets are considered in our research, together with an in-depth understanding of kinds of domain-knowledge.

2.4 Application domains

Now, we have contextualised our research and identified the problems to be addressed, in this section we describe the two domains that have been used to apply our proposals . These are the Emergency Response (ER) and the medical domains.

2.4.1 Emergency Response Scenarios

Emergency situations are generally referred to as disaster or crisis scenarios, caused by different factors: natural (e.g. biological, meteorological or geological), human-made (e.g. terrorist) or technological [115]. The complex and dynamic nature of information in these scenarios are examples of challenges confronted in situations in which the access to data and the understanding of different KR resources are limited. The complexity comes from different causes such as: the diversity of KRs between agencies' resources, the breadth of causes that can trigger responses and planning, or the lack of information at the first stages of emergency situations and the constrained access to specific agencies' sources, among others factors.

The reason for describing information as dynamic is because it is continuously changing, and in this particular case, it varies as the emergency situation advances. Therefore, the information that emergency agencies will have at the end of a crisis will be different than what they had when the emergency was triggered. This is why, it is absolutely crucial for agencies to have trusted and updated information at each point of the ER scenario. These aspects, along with that there are lives at stake, puts pressure on decision makers, depending, for the most part of their work, on the ability and willingness of the workforce [12]. Due to the relevancy and the impact of these decisions, researchers are focusing their efforts on proposing approaches and tools that help practitioners in the decision making process. Thus, Moßgraber et al. define a warning system architecture that includes multiple sensors to obtain relevant data, with the aim of anticipating disasters and starting responses as soon as possible [108]. In the same way, there are proposals that, by applying machine learning techniques to data from previous scenarios, try to discover relevant information of ER situations in order to optimise resources in future emergencies [112, 151].

In these scenarios it is essential that multiple agencies work together and share information, because the data of one agency might be relevant for the other organizations. For example, after the Fukushima Nuclear disaster, the access to the affected area was limited to mobile rescue robots, which gathered useful information and shared it with all agencies [109].

However, the automatic exchange of information is not trivial because it is usual that each agency use their own terminology and structure to store data, making difficult the collaboration among agencies. For this reason, researchers remark the need of defining a shared communication platform, in order to tackle this problem [125]. Establishing such a platform is challenging due to the diversity of known and unknown organizations with various core missions and perspectives, different levels of trust in these organizations and their data sources [103], complex access policies when private and confidential data are involved, and particularities in terms of natural and artificial representations and technological implementations [105]. Moreover, such platform only might work when a fixed number of agencies regularly collaborate, as they may have pre-alignments of their resources prior to the emergency, but ER scenarios are likely to entail the participation of new agencies, so their resources need to be aligned in the least possible time or even on-the-fly during the response. In this way, Segev proposed a method to modify ontologies in real time during crisis scenarios [127], however the application of this approach is complex because of the constraints of each organization.

Regarding ontologies in ER scenarios, we can refer to the Human Assistance Ontology [78, 127], which is attempting to put into effect an approach to ontology development in the humanitarian response field. Nevertheless, this is not trivial because it is assumed that every practitioner can interpret and use the ontology properly and sometimes due to continuous changing and access constraints, practitioners cannot understand and maintain that ontology, which in some cases may not represent the knowledge that they need. This makes that some agencies reject using such generic ontology and keep working with their digital resources which cover all their needs.

A common shared vocabulary between ER organisations would be really useful and beneficial to allow information exchange, however, this is not realistic because the KR of each agency evolves with a different pace and it is unrealistic that every single term used by ER agencies could be included in an agreed common terminology. Apart from that, as it was pointed out above, collaborations between some organisations or institutions might only be necessary in extraordinary situations, which means that these agencies do not exchange information frequently, and so their terminologies are not known to each other. In addition, ER gathers participants from many different areas of expertise with really diverse knowledge, which means that in some cases the exchange of information between two agencies might seem a priori unnecessary (e.g. the power supplier agency and the ambulance service). For this reason, it is not advisable that all agencies share an unmanageable shared vocabulary which includes every single term used by them to represented the knowledge that they need.

Unlike the mentioned approaches, McNeill et al. proposed the Combining Heterogeneous Agencies' Information (CHAIn) system [105]. This is a query answering system that carries out query rewriting in order to retrieve information that has been stored in agencies' systems with different terms, but representing the same knowledge. For example, *agency A* may store the data about the roads of Scotland in a database with a table called *road* whereas *agency B* may store the same information within a table called *route*. Thus, CHAIn will perform both queries[3]:

```
SELECT * FROM road
SELECT * FROM route
```

Therefore, if *agency C* asks for information of either *roads* or *routes* in the county of Midlothian, CHAIn will retrieve all the corresponding data stored in tables: *road* (*agency A*), and *route* (*agency B*).

CHAIn carries out the query rewriting process by applying OM on-the-fly, that is, performing OM with no previous alignments between agencies' resources [137]. The idea behind using OM is that each query has a structure that could match with one part of the structure of the agencies' resources. Following the

[3]The queries of the example are expressed in Structured Query Language (SQL) for simplicity. Nonetheless, CHAIn performs SPARQL Protocol and RDF Query Language (SPARQL) queries in order to exploit resources developed in RDF.

previous example, if *agency C* queries "*road*", the system will attempt to match *road* with an element conceptually equivalent to *road* within the structure of the agencies' resources. To do so, CHAIn uses the Structure-Preserving Semantic Matching (SPSM) [56, 104] algorithm which runs on top of S-Match [58]. In the example, the structure of the query is easy because it only contains one word, but the more complex the query is, the more complex its structure will be, and so, it is less likely to find a matching with the structure of a resource.

Knowing how CHAIn works, it can be considered as a suitable system for a first approximation to tackle data sharing heterogeneity problems in ER scenarios because the query rewriting functionality is a really useful way of retrieving data that represents the same knowledge, but diversely. Indeed, this solution is really interesting because its intention, far from being an autonomous or an unsupervised system, is to assist practitioners in the decision making process by providing the information that they are demanding. However, so far, the use of CHAIn in real ER scenarios is not optimal, because the system's matcher only has a domain-independent KB which does not include knowledge about ER scenarios. Thus, it is necessary to enrich the KB with ER domain-knowledge, as well as adding DA functionality in order to reach an acceptable performance of the system, and so, be apt to be used in real ER scenarios.

The application of our research to this system is straightforward because it uses S-Match and we are adding domain-knowledge to this matcher for our experiments. However, CHAIn mainly focuses on ER scenarios, where on-the-fly OM is required. That means, that its application to other domains in which matching can be done offline (e.g. the medical domain) is not justified. The lack of ER resources to evaluate our approaches, has entailed that we decided to consider the integration of our research within CHAIn as future work (see Section 7.2.5).

2.4.2 Medical Domain

Health is an area of knowledge which gathers professionals with different kinds of expertise. For example, within a hospital we can identify not only clinical workers (e.g. physicians, nurses, therapists, psychologists, pharmacists...), but also non-clinical ones (e.g. social workers, human resources, administrative assistants...). The collaboration between this mixture of professionals is essential for the hospital to give patients the best possible medical assistance. Therefore, working in a hospital implies interacting and coordinating with people from other areas of expertise. An example might be the bidirectional communication between physicians and nurses. In this case, the former communicate with nurses to indicate, for example, the treatment of a patient whereas the latter inform physicians of patient's evolution (e.g. any anomalous symptom or significant change in his/her vital signs since last medical examination). In this example, we can see how both, physicians and nurses, need to share common knowledge that allows them to understand each other, and so, perform effective communication [92]. For this reason, with the aim of avoiding communicating problems within the health environment, it is not unusual that the responsible institution of the hospital publishes a resource that compiles the "*official*" definitions of some of the most commonly used terms (e.g. a glossary of terms). This practise of generating and using their own knowledge resources, solves communication problems within each particular health environment. However, communications between professionals from different hospitals outside this environment can suffer from heterogeneity problems because the majority of these resources are circumscribed to specific health environments. Common examples are acronyms, which depending on the hospital, might not be defined, or defined, but with different meanings. For example, comparing two glossaries of terms one from South West London hospitals [33] and the other from hospitals in Sheffield [63], we can find some terms that overlap, but many others that are particular to each institution. Below there are some examples of acronyms defined in the mentioned glossaries:

- Acronyms shared in both glossaries:
  - *A&E*: Accident & Emergency.
  - *COPD*: Chronic obstructive pulmonary disease.
- Acronyms specific to each glossary:
  - South West London glossary:
    - *EIP*: Early intervention in psychosis.
    - *SGH*: St. George's Healthcare NHS Trust.
  - Sheffield glossary:

* *CVD*: Cardiovascular disease.
* *STH*: Sheffield Children's Hospital.

Similarly as in the previous scenario, where hospitals generate their "*official*" resources, another case that produces heterogeneity problems is the coexistence of several standard medical resources, which are daily used by health institutions. The two most popular standards are the Systematized Nomenclature of Medicine Clinical Terms (SNOMED-CT) [37] and the International Classification of Diseases (ICD) [116]. Although each resource has been developed for pursuing a different purpose (the former aims at capturing and representing patient data whereas the latter is used to assign diagnostic [23, 138]) in real-life, depending on the country, region or hospital, health professionals use SNOMED-CT or ICD for encoding diagnostics. The medical field knows the importance of having alignments between these two standards, so they regularly (twice per year) release a file with the mappings between SNOMED-CT and ICD-10. These mappings are statistically generated, so it is not unusual to find incorrect mappings, which are amended in following releases.

Apart from the differences of knowledge representation that can be found between SNOMED-CT and ICD-10, there are also different versions and adaptations of these resources. This fact, adds complexity to the data sharing process because two hospitals might experience heterogeneity problems even using the same standard, because they may have different versions.

In the case of ICD the appearance of heterogeneity problems is less likely because the migration from the International Classification of Diseases, version 9 (ICD-9-CM) to the International Classification of Diseases, version 10 (ICD-10) was carried out in the decade of 1990, so rarely there are hospitals which currently use this standard. However, all medical records before the migration were encoded using ICD-9-CM, so there are cases in which is required the mapping between both versions (e.g. in the generation of a dataset that contains health records of the last 30 years). Apart from that, ICD-10 has two adaptations, one published in Australia [46] (also used in New Zealand) and another introduced in Canada [115]. This means that each adaptation may have its own content, which may not be included in the ICD-10 official version, and so produces heterogeneity problems, for example, between an Australian hospital which uses the Australian ICD-10, and an American hospital that uses the official ICD-10 version.

Regarding SNOMED-CT, we can identify two major editions, the International Edition that contains the medical terminology in English, and the Spanish edition which is a translation of the International one [77]. Nonetheless, apart from these two main editions, there are also 9 national extensions. These extensions are from Australia, Canada, Denmark, the Netherlands, Spain, Sweden, UK, the United States (US) and Uruguay [77, 113]. Thus, the US SNOMED-CT has the medical terms from the International Edition plus specific terms. That means that exchange of information between hospitals from different countries, will encounter many problems when they try to understand information that is exclusively specified in the national extensions.

As we have seen with both medical resources, ICD-10 and SNOMED-CT, the national adaptations or extensions add complexity when it is necessary to exchange of information between medical experts across countries. Examples are the integration problems experienced by Robertson et al. when they tried to merge clinical data from different jurisdictions [121].

Although SNOMED-CT and ICD-10 cover most of terms and descriptions used in the different medical areas, there are cases in which medical specialists define a resource that exclusively includes knowledge of their medical specialisation. An example is DSM-5 [10], which is the reference manual used for mental health by psychiatrists and psychologists.

This kind of specialised resources within a domain, also produces communicating problems between physicians. For example, general practitioners normally use SNOMED-CT or ICD-10, whereas psychiatrists work with DSM-5. Thus, when a psychiatrist generates a health record for a patient, the information will be written using DSM-5 descriptions which are more detailed than the ones included in ICD-10 and used by general practitioners. In order to facilitate communication between physicians, DSM-5 includes mappings between the description of the diseases that it contains, and the diseases defined in ICD-10. Obviously, the degree of detail and accuracy is deeper in DSM-5 than in ICD-10, but at least these mappings allow that both, general and specialised medical experts, can communicate effectively.

The significant increase of medical resources, has motivated the idea of developing a system that brings these resources together to enable interoperability between computer systems. As a result, it was released the UMLS [20], which is a set of files and software that integrates health and biomedical resources.

Currently, UMLS is the reference system in the biomedical community because it contains dozens of biomedical resources, including the most important ones. However, far from reaching a consensus of representation between resources, the system just integrates all of them and makes them available. Thus, if one term appears in several resources, but having different meanings, the system includes all the meanings as well as the provenance resource. UMLS has three tools or knowledge resources:

- *Metathesaurus*. Compendia of terms and codes from many vocabularies including ICD-10, SNOMED-CT, Medical Subject Headings (MeSH), among others.
- *Semantic Network*. It contains a set of categories and the semantic relationships between them.
- *SPECIALIST Lexicon*. General English lexicon with biomedical terms.

Considering the previous analysis of the heterogeneity problems that take place because of the diversity of resources within the medical domain, we conclude that there is a real need of applying OM techniques to tackle these problems. However, it is necessary to highlight that the needs of the matching problem in the medical domain differ from the ER domain. Thus, whereas in the latter the matching process needs to be done on-the-fly, in the former it is possible to carry out the matching process offline.

The OM community is really interested in applying their techniques in the medical domain [73, 128], and evidences of this is that different biomedical ontologies are included every year in the OAEI in order to evaluate OM approaches [21, 80, 74].

In our case, we focus on applying our approaches to match ICD-10 and DSM-5 because we consider really challenging and interesting the task of matching two resources with different degrees of specialisation, as there might not be a priori a clear matching between the elements of the different resources. Both, ICD-10 and DSM-5 are described in more detail in Section 3.4.

2.5 Summary

In this chapter, we reviewed some milestones from the beginning of the KR research area until the need of applying OM techniques in order to deal with diverse knowledge representations when sharing information between different ontologies. In this review, we have identified different problems produced because of the lack of domain knowledge in matchers' KBs. In our research, we focus on reducing the number of false negatives and false positives, instantiating our approach to the ER and the medical domains, even though due to the lack of accessible ER resources we have evaluated our approaches exclusively with medical resources. The particularities of each domain and the interest of applying OM techniques in these domains were addressed at the end of this chapter.

Chapter 3

Background

In this chapter, we introduce several concepts, tools and resources that are key to understand our approach and its evaluation. First of all, the matching process and relevant terminology are described. Secondly, we describe the matchers that have been used in our research. After that, there are presented the tool and the general knowledge resource used to integrate domain-knowledge into matchers' KB, as well as the domain-knowledge resources that have been used. The chapter concludes with the two classifications of diseases that are used in the evaluation (Chapter 6).

3.1 Ontology Matching

In chapter 2, we have seen how OM is required because of the need of addressing problems caused by resources with diverse knowledge representations. We can find in the literature many contributions presenting different solutions to tackle these problems, each one with its particularities (algorithms, similarity measures, parameters...). Nonetheless, all of them have in common some terms which refer to the process and the elements involved in it. Despite this, there is no consensus on the meaning of these terms, it being not unusual to find them with different meanings. For this reason, it becomes necessary that we define some of these essential concepts, in order to avoid any trace of misunderstanding. In our case, we follow the definitions developed by Euzenat and Shvaiko in [38], because this is the reference book of OM, and so it is widely accepted by the researchers of this community.

3.1.1 Basic Concepts

The *matching* task is a process in which for a given pair of ontologies $O1$ and $O2$, it is found an *alignment* A'. This basic definition can be extended by the use of: i) an input alignment A, which will be extended with the new alignments discovered in the matching process, ii) specific parameters such as weights or thresholds, and iii) external resources that provide BK.

Figure 3.1 depicts a general schema of a matching process.

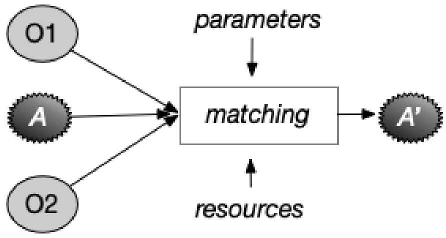

Figure 3.1: Matching process schema (adapted from [38])

An *alignment* is a set of correspondences between the entities of the ontologies used as input in the matching process. Therefore, it is the output of this process. Depending on the input ontologies, alignments may have different cardinalities: 1:1 (one to one), 1:m (one to many), m:1 (many to one) or m:m (many to many). In our research we have found mostly cardinalities 1:1 (e.g. *command and control levels*: *gold-strategic*, *silver-tactical*, *bronze-operational*) and 1:m (e.g. *evacuation*-[*mass evacuation, medical evacuation, local evacuation*]) (see figure 3.2).

A *correspondence* can be defined as a 4-tuple:

$$< id, e_1, e_2, r >$$

where:

- id is the identifier of the given correspondence;
- e_1 and e_2 are entities of $O1$ and $O2$;
- r is a relation between e_1 and e_2.

Therefore, the correspondence $< id, e_1, e_2, r >$ asserts that the entities e_1 and e_2 hold the relation r between them. Figure 3.2 shows different correspondences between $O1$ and $O2$. If we focus on *Evacuation*$(O1)$

and *Medical Evacuation*$(O2)$ we can see the following correspondence $< id_{1,1.1}, Evacuation, MedicalEvacuation, \sqsupseteq >$ which asserts that *Evacuation* in $O1$ is more general ($\sqsupseteq$) than *Medical Evacuation* in $O2$.

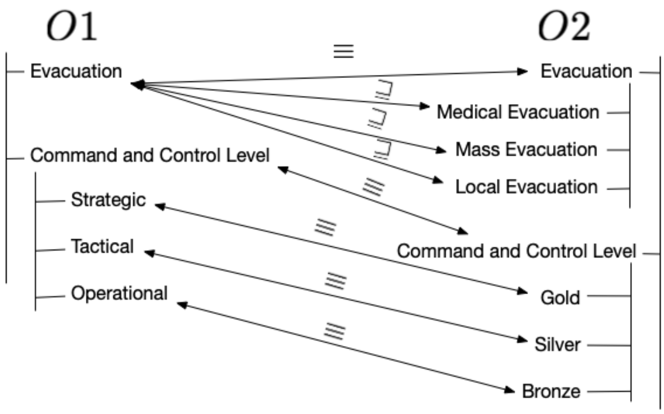

Figure 3.2: Alignment between ontologies $O1$ and $O2$.

There are matchers that enrich correspondences with *metadata*. A common metadata is a value which indicates the confidence of the correspondence. Thus, the higher the confidence is, the more likely is that the involved entities hold the relation specified in the correspondence.

Another term that needs to be defined is *mapping*. It is the oriented version of an alignment, in which the entities of one ontology are mapped to at most one entity of another ontology. Thus, it can be seen as a collection of rules all oriented in the same direction, which map an entity of one ontology into another one from another ontology [38].

3.1.2 Matchers

Once there have been defined some key concepts of OM. Below we describe the two matchers that are used in our research. The main reason for using these two matchers, apart from their relevance in the OM area, is because even though, both use BK, they perform the matching process differently. This is useful for us because we can analyse the behaviour of our approach in each case, synthesising the background of our proposal and making it extensible to other matchers that use BK.

3.1.2.1 S-Match

S-Match is a semantic matcher, which operates at the level of meanings of labels. It uses a BK that contains concepts linked by semantic relations. Therefore, the matcher takes advantage of the semantic information implicitly or explicitly codified in the input ontologies or schemas. Thus, S-Match can discover a mapping between two entities, even though, this mapping is not explicitly represented in the BK (e.g. there is no *synonymy* relationship between the entities). In particular, it focusses on the labels of nodes and arcs between nodes [58]. One of the main characteristics of this matcher is that it is highly precise in terms of relationships between the elements of the input schemas. Therefore, instead of simply identify a certain degree of similarity between these elements, it also differentiates between *equivalence* and *subsumption* relations. Figure 3.2 shows examples of the relations that are recognised by S-Match.

The matching process consists of the following main steps [58]:

1. **Preprocessing**
   Firstly, it is necessary to translate the input labels which are represented in an "*external language*", such as natural language, into the language in which the concepts are expressed in the matcher (*internal language*). In this case, S-Match has an *internal language* which defines the syntax and semantics, using propositional logic, where atomic formulae are atomic concepts, written as single words, and complex formulae are obtained by combining atomic concepts using the connectives of set theory.

   (a) **Computing concepts (C_L) for all labels in both trees.** The translation process from labels to concepts has the following steps:

      i. *Tokenization.* Every label identified in each node is parsed into tokens. For example, "*depersonalization derealization syndrome*", becomes <*depersonalization, derealization, syndrome*>.

      ii. *Lemmatization.* Each token discovered in the previous step is lemmatized, being morphologically analysed in order to find its possible basic forms. For example, *doctors* is associated with its singular form *doctor*.

      iii. *Building atomic concepts.* S-Match's KB (WN) is queried to extract the sense of the lemmas identified in the previous step.

For example, *doctor* has 7 different senses [44], 4 are nouns and 3 are verbs.

Nouns:

- A licensed medical practitioner.
- (Roman Catholic Church) a title conferred on 33 saints who distinguished themselves through the orthodoxy of their theological teaching.
- Children take the roles of physician or patient or nurse and pretend they are at the physician's office.
- A person who holds Ph.D. degree (or the equivalent) from an academic institution.

Verbs:

- Alter and make impure, as with the intention to deceive.
- Give medical treatment to.
- Restore by replacing a part or putting together what is torn or broken.

S-Match uses the *sense filtering* heuristic method [22, 94] to choose the most likely sense by considering the context. For example, the entity *doctor* within a hospital ontology, will have the first sense.

iv. *Building complex concepts.* Tokens such as prepositions, punctuation marks and conjunctions are translated into logical connectives. After that, they are used to build complex concepts using the atomic ones, identified in the previous step.

After these steps, all labels have been translated into sentences of the internal concept language.

(b) **Computing concepts (C_N) for all nodes in both trees.** These are written using the same internal language as concepts of labels. The main idea here is that the concept C_n of node n, is computed as the intersection of the concepts at labels of all the nodes from the root to the node itself.

2. **Computing relations among (C_L) for all pairs of labels in the two trees.** The C_L matrix, which contains the relations between any two concepts of labels in the two trees, is computed in this step. At this point a

lot of prior knowledge is required. S-Match uses two sources of information [57]:

(a) "*Weak semantics element level matchers*". These algorithms carry out string manipulation in order to guess semantic relations implicitly encoded in similar words. For example *REM* and *rapid eye movement*. It is necessary to highlight that these matchers return a semantic relation rather than a [0,1] affinity level. For instance, in the previous example the matchers returns a *synonymy* relation as *REM* is an acronym of *rapid eye movement*.

(b) "*Strong semantics element level matchers*". These algorithms extract semantic relations existing between concepts of labels by using oracles which have the necessary lexical and domain knowledge. Examples of oracles might be WNs, domain ontologies or thesauri among others. The possible semantic relations are:

- *Equivalence* ($\equiv$). The labels are equivalent. E.g. *illness* $\equiv$ *sickness*.
- *More general* ($\sqsupseteq$). One label is more general than the other label. E.g. *disease* $\sqsupseteq$ *rheumatism*.
- *Less general* ($\sqsubseteq$). One label is less general than the other label. E.g. *rheumatism* $\sqsubseteq$ *disease*.
- *Mismatch* ($\perp$). The labels are antonyms or represent different senses of the same set of concepts. E.g. *ill* $\perp$ *well*.

3. **Computing relations among (C_N) for all pairs of nodes in the two trees.** The C_L matrix computed in the previous step contains a lot of BK codified as semantic relations between concepts of labels of the two trees. The main idea of this step is to take advantage of this BK which provides the context with which the matcher reasons [52]. Thus, it is necessary to translate all the semantic relations into propositional connectives. That is: equivalence into equivalence, more general and less general into implication, mismatch into negation of the conjunction. So, then we can prove that:

$$Context \rightarrow rel(C_i, C_j)$$

is valid; where C_i is the concept of node i in the graph 1, C_j is the concept of the node j in graph 2, rel is the semantic relation that we want to prove holding between C_i and C_j, and *Context* is the conjunction of all the relations between concepts of labels mentioned in C_i and C_j.

The integration of domain-knowledge in S-Match's KB, affects directly steps 1.a.iii (*building atomic concepts*) and step 2 (*compute relations among (C_L) for all pairs of labels in the two trees*).

3.1.2.2 LogMap

As opposed to S-Match, LogMap cannot distinguish between different semantic relations. It only identifies a mapping when two elements achieve certain degree of similarity above a predefined threshold. At this point, the identified mappings are considered as *equivalences*.

Currently, LogMap is one of the matchers which performs better in terms of speed, precision and recall in the OAEI [4, 5, 74]. The main reason for such good performance is because LogMap is designed as a highly scalable ontology matching system, implementing data structures that are highly optimised for indexing lexically and structurally input ontologies, so that it can easily process all the large datasets proposed in the OAEI. The matcher also has reasoning and diagnosis capabilities which are used to refine mappings in an iterative process, until it outputs the final alignment.

In broad strokes, LogMap computes an initial set of mappings (*anchor mappings*), assigning a *confidence* value to each of them. After that, an iterative process is carried out, which alternates *mapping repair* and *mapping discovery* steps [79]. Figure 3.3 depicts the matching process performed by LogMap.

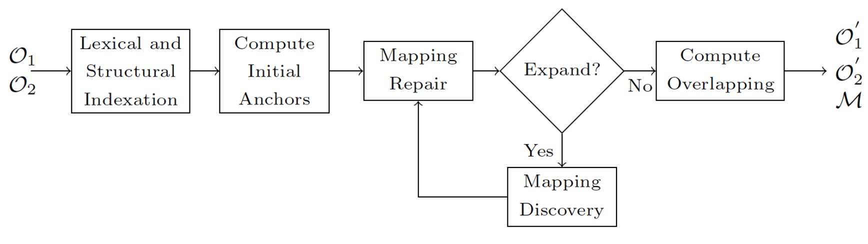

Figure 3.3: LogMap in a nutshell (extracted from [79])

The process is explained in detail as follows in [79]:

1. **Lexical indexation.** LogMap assigns a unique index to each label of the classes in each ontology, as well as their lexical variations. Additionally,

it is possible to enrich the indexes by using an external lexicon. The indexation process is commonly used in IR, allowing to efficiently access the information associated with an index.

2. **Structural indexation.** The matcher uses an internal labelling schema [110] to represent the extended class hierarchy of each input ontology. Doing that, LogMap is able to have efficient access to the information in the hierarchy even when representing large ontologies. Moreover, LogMap allows to compute each extended hierarchy by using either simple structural heuristics or an off-the-shelf description logic reasoner.

3. **Computation of initial 'anchor mappings'.** LogMap computes an initial set of equivalence *anchor mappings* by intersecting the lexical indexes of each input ontology. These mappings can be considered 'exact' and will later serve as starting point for the further discovery of additional mappings. For each anchor, LogMap assigns a confidence value based on context similarity, using the ISUB tool [133]. Thus, given an anchor, $m = (C1 \equiv C2)$, the ISUB tool computes the confidence of m considering the principle of locality. That is, if the hierarchy neighbours of $C1$ in $O1$ and $C2$ in $O2$, match with low confidence, then the anchor may be incorrect. For example, the class *Trapezoid* in two different ontologies ($O1$ and $O2$) can be an anchor mapping as they are homonyms and have high string similarity. For this reason, LogMap includes $m = Trapezoid(O1) \equiv Trapezoid(O2)$. However, if *Trapezoid* is classified in $O1$ as a *polygon*, and as a *bone* in $O2$, LogMap will assign low confidence to this anchor mapping (m), making it likely to be eliminated from the alignment in the repair step.

4. **Mapping repair and discovery.** The core of LogMap is an iterative process that alternates *repair* and *discovery* steps.

   - In the *repair step*, a reasoning algorithm is used to detect classes which are unsatisfiable w.r.t. both input ontologies and the mappings computed thus far. Then, each of these undesirable logical consequences is automatically repaired using a 'greedy' diagnosis algorithm.
   - To *discover new mappings*, LogMap maintains two *contexts* (sets of 'semantically related' classes) for each anchor. Contexts for the same

anchor are expanded in parallel using the class hierarchies of the input ontologies. New mappings are then computed by matching the classes in the relevant contexts using ISUB. This mapping discovery strategy is base on a *principle of locality*: if classes C_1 and C_2 are correctly mapped, then the classes semantically related to C_1 in O_1 are likely to be mapped to those semantically related to C_2 in O_2. LogMap continues the iteration of repair and discovery steps until no context is expanded in the discovery step. The output of this process is a set of mappings that are likely to be 'clean' - that is, it will not lead to logical errors when merged with the input ontologies.

5. **Ontology overlapping estimation.** LogMap computes a fragment of each input ontology, which represent the overlap between them.

In LogMap, the integration of terms affects the first step (*lexical indexation*), so this is the step on which we focus when applying our approaches.

3.2 Integration tool - Diversicon

Apart from the described matchers, the keystone of our research is the tool that is used to integrate domain-specific knowledge into matchers' BK. In particular, we have used Diversicon to which we have contributed with our domain-specific extensions and with the integration of LogMap.

The term *extension* is used in the thesis to refer to a domain-specific knowledge resource that has been developed from a source knowledge resource (e.g. MeSH) and that can be integrated into a domain-independent KB (in our case, we focus on WN).

Diversicon [15] is an Open Source Framework (see figure 3.4) that extends the UBY framework [70], which is a large-scale lexical-semantic resource for NLP based on the LMF standard [47, 48].

As we can see in figure 3.4, Diversicon has four different parts:

- *Static resources.* It contains different knowledge resources such as WNs and domain-specific resources. These resources are available in a online catalogue called the *Diversicon Catalogue*[1]. The catalogue includes the ER and medical extensions developed during our research.

[1] http://diversicon-kb.eu

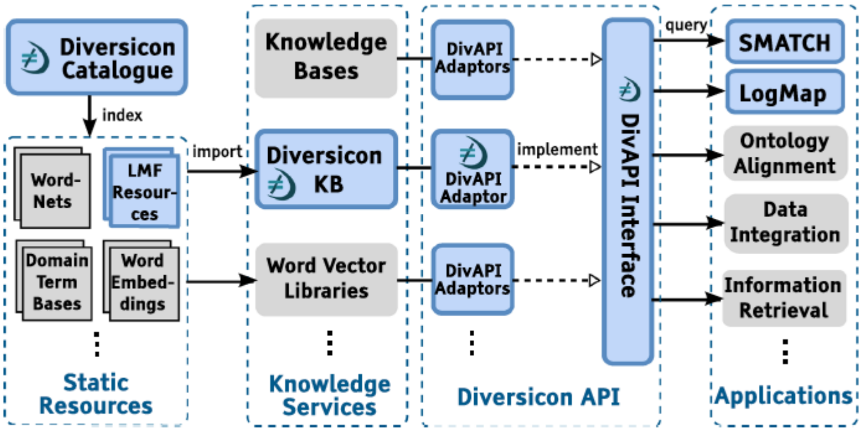

Figure 3.4: High-level architecture of Diversicon (extracted from [15])

- *Knowledge services.* The main functionality of Diversicon is that it can be used to integrate and access lexical domain-knowledge in a unified manner. The use of the LMF standard makes easier the integration of resources into WN, which is developed using the same standard. Thus, it is possible to create personalised knowledge resources, by combining existing ones. By default, Diversicon imports WN which contains domain-independent knowledge to which it is possible to integrate more specific knowledge. In our case we have generated two different KBs, with general knowledge plus ER and *medical* knowledge respectively.

- *Diversicon Application Programming Interface (API).* These are different adaptors that allow Diversicon to be integrated with third-party tools. Using the API it is possible to carry out all the tasks of the knowledge services described before, which are accessible by the client called Divercli[2]. In our case, we have used them in order to allow S-Match and LogMap accessing to an enriched KB.

- *Applications.* These are the applications that are compatible with Diversicon. In particular, in our case we have used S-Match and LogMap. It is necessary to remark that these applications are not included in the framework, but they can be used with the *Diversicon API.*

It is necessary to highlight that Diversicon has had an essential role in our research because it has allowed us to enrich matchers' KB with domain-specific

[2]http://diversicon-kb.eu/manual/divercli/

knowledge resources. Moreover, all of these resources produced during the thesis are included in the Diversicon Catalogue, so anyone can take advantage of them. Therefore, apart from playing a crucial role during the thesis, this tool will help to disseminate our research and make it available to the OM, biomedical and ER communities.

3.3 Knowledge Representation Resources

In this section, there are described the knowledge resources that we have used during our research as matcher's BK. These are four resources in total: one domain-independent lexical resource, and three domain-specific resources (one from the ER domain and two from the medical domain).

3.3.1 WordNet

WN [44] is one of the most extensive domain-independent lexical resources. It is organised as a Directed Acyclic Graph (DAG), which describes semantic relationships between sets of synonyms, called synset. Thus, words are grouped into synsets. For example, *illness* and *sickness* are two words which belong to the same synset.

Regarding the semantics relations, apart from the *synonymy* and *antonymy* relations, WN also has the following ones [134, 145]:

- *Class inclusion (is-a).* This relationship allows synsets to be organised from more general to more specific. Two semantic relations derive from it:
  - *Hypernymy.* One synset is more general or a superconcept of another synset. For example, *illness* is the hypernym of *anuresis* ($illness \sqsupseteq anuresis$).
  - *Hyponymy.* A synset is more specific or a subconcept of another synset. For example, *anuresis* is a hyponym of *illness*. ($anuresis \sqsubseteq illness$).
- *Meronimic inclusion (part-of).* This relationship occurs between a synset and its parts. Two semantic relations derive from it:
  - *Meronymy.* One synset is part of another one. For example, *eye* is a meronym of *face*.

– *Holonymy*. A synset has/includes another one. For example, *face* is a holonym of *eye*.

Currently, matchers take advantage of the *is-a* relationship, because using it, they can infer *subsumption* [58]. However, they do not use the *part-of* relationship, because even though this semantic relation gives information that relates concepts, that information is not enough to conclude that there is a mapping between two concepts that have a *part-of* relationship. For example, *eye* cannot be mapped with *face*.

Apart from the mentioned semantic relations, WN also includes other relations such as: *similar to, attribute*, or *derivationally related form*. In our case, we also focus on the last one, which indicates that a lexical entry is a derivation from another lexical entry. For example, from *virus* (noun), a derivation is *viral* (adj). This relation is essential to us because using it we carry out morphological expansion of domain-specific knowledge (see section 4.4.2.1).

One of the main reasons of why WN is widely used is because it represents general knowledge from a wide variety of areas of expertise. It includes a huge number of nouns, adjectives and verbs. There are more than 155,000 lexical entries organised in more than 117,000 synsets with more than 207,000 word-sense pairs [44].

In our case, we use WN as a basis of general knowledge into which we plug domain-knowledge. Specifically, we have extended it with knowledge from the ER and medical domains. For the purpose of our research we have generated two versions of enriched WN, one with the ER extension and another with the medical extensions. However, it is totally possible to have only one version of WN enriched with different extensions of domain-knowledge.

3.3.2 Medical Domain

Regarding the medical domain, we have worked with two of the most used resources in this domain, the MeSH and the SPECIALIST Lexicon (SPECIALIST).

3.3.2.1 MeSH

MeSH is the National Library of Medicine's controlled vocabulary thesaurus[3]. It consists of sets of terms naming descriptions in a hierarchical structure that permits searching at various levels of specificity.

MeSH descriptors are arranged in both an alphabetic and hierarchical structure. At the most general level of the hierarchical structure there are very broad headings such as "*Anatomy*" or "*Mental Disorders*". More specific headings are found at more narrow levels of the thirteen-level hierarchy, such as "*Ankle*" and "*Conduct Disorder*". There are over 28,000 descriptors in MeSH with over 90,000 entry terms that assist in finding the most appropriate MeSH headings, for example, "*Vitamin C*" is an entry term of "*Ascorbic Acid*". In addition to these headings, there are more than 240,000 Supplementary Concept Records (SCRs) within separate files. Generally SCRs contain specific examples of chemicals, diseases, and drug protocols. They are updated more frequently than descriptors. Each SCR is assigned to a related descriptor via the Heading Map (HM) field, which is used to rapidly identify the most specific descriptor class and make it accessible.

The main headings in MeSH are listed by a tree number system that places the headings in a hierarchical arrangement. This hierarchy has relations such as *part_of* (see figure 3.5) and *is_a* (see figure 3.6).

A01.456..Head
A01.456.313..Ear
A01.456.505..Face
A01.456.505.173..Cheek
A01.456.505.259..Chin
A01.456.505.420..Eye
A01.456.505.420.338..Eyebrows
A01.456.505.420.504..Eyelids
A01.456.505.420.504.421..Eyelashes

Figure 3.5: Example of part_of relation

A02.513..Ligaments
A02.513.170..Broad Ligament
A02.513.514..Ligaments, Articular
A02.513.514.100..Anterior Cruciate Ligament
A02.513.514.162..Collateral Ligaments
A02.513.514.162.250..Collateral Ligament, Ulnar
A02.513.514.162.500..Lateral Ligament, Ankle
A02.513.514.162.600..Medial Collateral Ligament, Knee
A02.513.514.287..Ligamentum Flavum
A02.513.514.350..Longitudinal Ligaments
A02.513.514.475..Patellar Ligament
A02.513.514.538..Plantar Plate
A02.513.514.600..Posterior Cruciate Ligament
A02.513.514.800..Round Ligament of Femur
A02.513.901..Round Ligaments
A02.513.901.500..Round Ligament of Femur
A02.513.901.750..Round Ligament of Liver
A02.513.901.875..Round Ligament of Uterus

Figure 3.6: Example of is_a relation

As we have seen above, MeSH implements the same semantic relations as WN.

[3] https://www.nlm.nih.gov/mesh/

This fact, facilitates the generation of extensions able to be integrated into WN. In particular, we have generated an extension with all medical headings about *Diseases* (category C in MeSH) and *Psychiatry and Psychology* (category F in MeSH).

3.3.2.2 SPECIALIST

SPECIALIST is a resource included in the UMLS. This system contains a set of files and software that brings together many health and biomedical vocabularies and standards to enable interoperability between computer systems.

The SPECIALIST lexicon is intended to be an English lexicon which contains biomedical terms, including both commonly occurring English words and biomedical vocabulary.

It is composed of lexical records which form a frame structure consisting of slots and fillers. Each lexical record has a *base* (slot whose filler indicates the base form), and optionally a set of *spelling variants* (slots to indicate spelling variants) or morphological derivation. For example, the lexical entry with base "*nephroprotective*" (adj) has as spelling variant: "*nephro-protective*", and as morphological derivation "*nephroprotectivity*" (noun).

Lexical entries are not divided into senses, so an *entry* represents a spelling-category pairing regardless of semantics.

In our case, we have used SPECIALIST for enriching matchers' BK with two purposes: lexical and morphological. For the former we used the SM.DB[4] file, which has lexical variations linked with their respective synonyms. This resource does not have semantic relations between them, but it can take advantage of a resource included in the UMLS that is called the Semantic Network. The Semantic Network contains a set of categories or semantic types in which all concepts represented in the UMLS can be categorised, and a set of relationships existing between the semantic types. For the latter, we use the DM.DB which is table of different derivations. In particular, we focus on the *suffixD* file, which includes grammar rules that take place within medical terms.

[4]https://lsg3.nlm.nih.gov/LexSysGroup/Projects/lexicon/2018/release/LEX/LEX_DB/SM.DB

3.3.3 Emergency Response

Concerning ER, we have used the United Kingdom Civil and Protection (UKCP) lexicon, because this is the resource used by ER agencies within the UK. This is because we started collaborating with the Scottish Government Resilience, and they use this resource.

3.3.3.1 UKCP

This lexicon is a collection of more than 725 terms used between UK ER agencies in ER scenarios. Each entry in the lexicon contains: the label of the term, denoted as the primary term; version of inclusion or revision; source and definition. Moreover, there are some terms that also have abbreviations or acronyms, notes on definition and the jurisdiction to which the term is restricted. There are some definitions which reference particular terms that have been defined in the lexicon. These terms are represented in bold to be distinguished from general terms. Table 3.1 shows an example of the terms: agency, air ambulance, medevac and responder as they appear in the UKCP lexicon. Specifically, the entries of the terms in this example come from two different sources: the Civil Contingencies Secretariat (CCS) and the Environment Agency (EA).

At the beginning of our research we focused on applying our approaches to CHAIn [105] in order to improve the query rewriting functionality. For this reason, we developed an extension of 100 of the most common terms used by ER agencies in the UK, extracted from the UKCP, [119, 118] as well as a taxonomy of domains participating in ER scenarios [120]. This extension was integrated into the matchers' KB, making the matcher capable of finding new matchings. Thus, for example, if an ER agency asks for information about *active evacuations*, the system would be able to perform, apart from the query that looks for the records within a table called *evacuation*, as many queries as different kinds of evacuations are represented in the matchers' BK (e.g. *medical evacuation, small evacuation, mass evacuation*, and *large scale evacuation*). Apart from this example that shows two agencies that represent knowledge with different degrees of specificity, the extension also allows the matcher to match cases in which agencies represent the same knowledge, but using different terms. Figure 3.7 shows an example where *agency A* represents the *command and control* levels using the terms *strategic, tactical and operational*, whereas *agency B* uses *gold, silver and bronze*, and the

Primary Term	Abbr.	Vers.	Source	Definition	Notes	Jurisdiction
Agency		2.0	CCS	A generic term, widely used as synonymous with organisation.		
Air ambulance		1.0	CCS	Aircraft (usually a helicopter) used primarily to transport medical or paramedical staff to the **site** of an **incident** or **emergency** and **casualties** to specialist trauma centres and/or **designated hospitals**		
Environment Agency	EA	2.0	EA	The Environment agency's role is to protect and enhance the environment as a whole in England and Wales. As part of this role, they are also a nuclear regulator in respect of controlling discharges to the environment. An executive non-departmental public body responsible to the Secretary of State for **DEFRA**.	Note: in Scotland, this role is performed by the SEPA and the Scottish Environment Protection Agency.	*England and Wales*
Medevac		1.0	CCS	Abbreviation for Medical Evacuation (of casualties by air)		
Responder		1.0	CCS	Organisation required to plan and prepare a response to an **emergency**.		

Table 3.1: Entries in the UKCP lexicon

matcher will output the correspondences between both agencies.

Despite our advances and interest in this application area, the difficulties that we experienced trying to get access to ER data, and so, to key resources for evaluating our approaches, entailed that we made the decision of considering the application of our research in ER scenarios as a future work to be addressed after

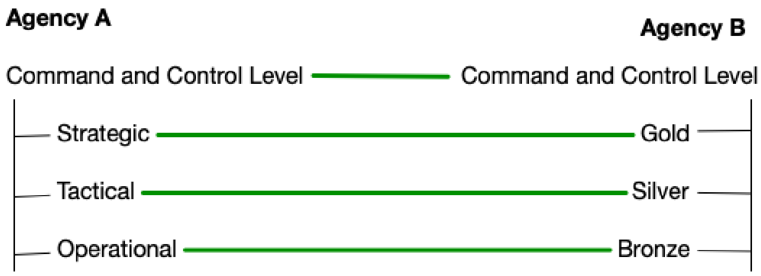

Figure 3.7: Example of mappings discovered by matchers enriched with the UKCP extension.

the PhD. For this reason, we decided to find another area of expertise which also present heterogeneity problems and that have accessible resources to evaluate our proposals. As a result, we opted for the medical domain.

3.4 Classifications of diseases

Apart from the knowledge resources used to enrich matchers' BK with general purpose and with domain-specific knowledge, we have also used two of the most popular classifications of diseases to evaluate our approaches. These classifications are ICD-10 and DSM-5.

3.4.1 ICD-10

ICD-10[5] [116] is the most extended classification of diseases around the world. An evidence of this are the initiatives and policies that the European Union is promulgating to persuade professionals of the health domain into encoding health records using ICD-10.

The classification is structured into a taxonomy with 22 chapters that contain blocks of kinds of diseases. Each chapter has several blocks of diseases following an *is_a* relationship. Figure 4.3 shows an example of how the disease "*Obesity*" is represented in ICD-10. We can see how this disease is included in category *IV - Endocrine, nutritional and metabolic diseases* and in the subcategory *E66.- Obesity*. Thus, physicians should specify the kind of obesity that a patient has, by assigning the specific code (e.g. *E66.1 Drug-induced obesity*).

As described in Section 3.4, currently, there are different editions of ICD-10.

[5] https://www.cdc.gov/nchs/icd/icd10cm.htm

Although the International edition defines the basic codification of diseases, different countries have adapted this resource to their needs, by adding new codes. In our case, we use the International edition because we are not restricting our experiments to a specific region.

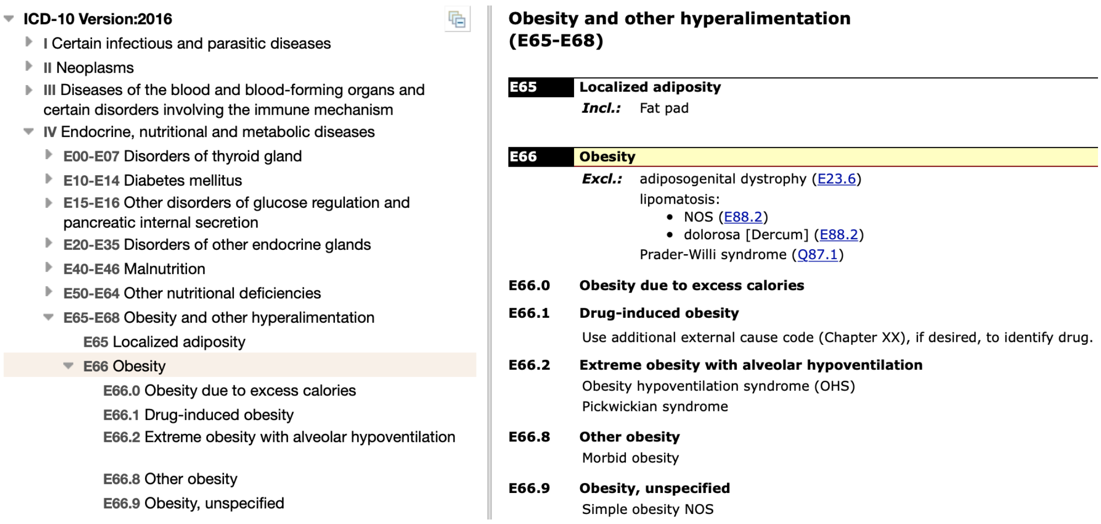

Figure 3.8: Representation of *obesity* in ICD-10

The resource has thousands of entries, to allow the codification of all known diseases. In our case we focus only on Chapter V "Mental and behavioural disorders (F00-F99[6])" because we carry out our experiments by matching ICD-10 with a resource that only contains descriptions of mental health diseases.

3.4.2 DSM-5

DSM-5[7] [10] is the reference manual of mental health used by psychiatrists and psychologists. It contains a classification of mental health disorders with descriptions, sorted in a table in which each entry has two fields: (i) the ICD-10-CM code, and (ii) the associated "Disorder, condition or problem".

Therefore, it provides a direct mapping between its "description" and the representation in ICD-10. In our case, we deliberately withheld these mappings so as to see if after applying our approach S-Match and LogMap could rediscover them in order to evaluate our hypothesis. For this reason, due to these mappings have been generated manually by experts in the field, we consider them as our gold standard to evaluate the performance of the matchers.

[6] http://apps.who.int/classifications/icd10/browse/2016/en#/V
[7] https://www.psychiatry.org/psychiatrists/practice/dsm

Table 3.2 depicts an extract of DSM-5 which includes the representation "*E66.9 - Overweight or obesity*".

ICD-10-CM	**Disorder, condition, or problem**
E66.9	Overweight or obesity
F01.50	Probable major vascular neurocognitive disorder, Without behavioral disturbance
F01.51	Probable major vascular neurocognitive disorder, With behavioral disturbance
F02.80	Major neurocognitive disorder due to another medical condition, Without behavioral disturbance
F02.80	Major neurocognitive disorder due to HIV infection, Without behavioral disturbance

Table 3.2: Example of the representation of *obesity* in DSM-5 (extracted from [10])

3.5 Summary

In this chapter, we briefly introduced some key concepts, resources and tools that are used in the thesis. Firstly, we described basic concepts of OM, and the matchers used for evaluation purposes. After that, we pointed out the tool used for plugging domain knowledge into matchers KB. Finally, we detailed the resources that have been plugged and the medical classification of diseases used for evaluating our approach.

Chapter 4

Enhancing Ontology Matching with Domain-Knowledge

This chapter contains the research contribution of the thesis. First of all, the proposal is contextualised by highlighting the particular problem of Natural Language Understanding (NLU) that we tackle in the OM process. After that, there are introduced the three dimensions of domain-knowledge that we have identified, and the methods and the resources fruit of their combinations. The chapter finalises with the architecture of our solution and several approaches to take advantage of these domain-knowledge dimensions in the OM processes.

4.1 Natural Language Understanding for Matching

The understanding of natural language is a topic in which researchers have been concentrating their efforts since last mid-century [7, 144]. NLU is of a particular interest in the OM area, where it can be used to carry out a process to convert the informal natural language of ontologies' labels into formal representations. This process of label formalisation (also called *idealisation* [27]) is an essential step of the OM process, because this provides the matcher with the representation of the ontology entities. However, it is important to remark that matchers do not need to completely understand the labels in order to be matched because they can infer the mappings (e.g. applying only string similarity measures). In our research, we consider a *label* as the string assigned to a node in an ontology, which may be composed of one or more words. It is necessary to highlight, that there are no standards or conventions that indicate the way in which ontology labels should be defined. Therefore, they usually have different formats in different ontologies, making it necessary to process them in a common format.

Mainly, matchers address the matching problem by using the following NLU approaches [38]:

- *String-based techniques.* They consider each ontology label as a sequence of characters. Therefore, it is possible to apply string similarity measures such as the Hamming's distance [72], the edit distance [19] or path comparison [59].

- *Language-based methods.* These methods rely on using NLP techniques to help to extract the meaningful terms from text. Basically, we can distinguish between *intrinsic* and *extrinsic methods.* The former methods, also called *language normalisation*, try to reduce each form of a term to a standarised form by carrying out tasks such as *tokenisation*, *lemmatisation*, *term extraction* or *stopword elimination.* The latter methods take advantage of external resources such as dictionaries, in order to find similarities between terms considering their meaning.

Generally, matchers combine both string-based techniques and language-based methods. However, when they find mappings by comparing ontology entities considering only their labels, they have a main problem caused by the existence of synonyms and homonyms, which may complicate the discovery of mappings and provoke ambiguity problems. An example might be the mapping of these entities with these two labels: *financial building* and *bank.* In some contexts, this mapping is right, however if *bank* belongs to an ontology from the geographical domain, the mapping will be wrong because *bank*, in this case refers to "*a slopy land beside a body of water*"[1].

Macro Components of Matchers

After analysing different matchers [58, 79, 117], we have defined a generalisation of the macro components that are part of a matcher, which is depicted in Figure 4.1.

Considering the matching process from the input of the ontologies until the output of the alignment, we identify two main processes :

[1] Definition extracted from WN.

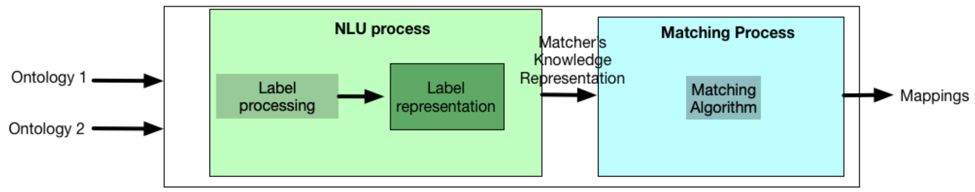

Figure 4.1: Matcher macro components

1. *NLU process.* Having the ontologies to be matched, this process consists of processing the input labels and representing them in the format of the matcher. To do so, there are applied both string-based and language-based methods.

2. *Matching process.* This process starts once the labels have been processed and represented in the matcher's format. To do so, the matcher uses the matching algorithm which discovers the mappings and generates the output with them.

The NLU step is crucial because if the labels are not well processed and represented, the matcher could not find the mappings. The complexity of this step increases when the ontologies have a high degree of specialisation because, normally, the external resources exploited by the matcher do not have represented such specialised knowledge, and so, it cannot represent the label. For example, if we try to match ontologies which represent medical terms such as *akathisia* or *frotteurism*, and the matcher does not use a resource which represents psychiatric knowledge, it is not possible to find the mapping between these entities.

Due to the need of using domain-knowledge resources for matching and the particularities that these resources have, it is necessary to carry out a thorough analysis of them. The main aim of doing this analysis is to find features that are common to any domain-knowledge resource, and that can be considered to address the mentioned OM problems. In our research, we analysed various resources from the ER and medical domains. From this analysis, we have identified three different dimensions or perspectives in which domain-knowledge can be considered: (i) *specificity*, (ii) *linguistic structure*, and (iii) *type of knowledge resource*.

The main goal of our research is the improvement of matcher's performance in domain-specific contexts by using various forms of domain-knowledge. Indeed, we focus on this subproblem of NLU in which our goal is the interpretation of labels in the ontologies to be matched, and by consequence the scope of the research is reduced to language-level domain-knowledge. Thus, our hypothesis is that *when matching domain ontologies, matchers with DA functionality have a better performance in terms of precision and recall than those which do not have this functionality because domain knowledge helps them to disambiguate and discover mappings that otherwise could not be found, and reject mismatches that look superficially plausible.*

4.2 Dimensions of Domain-Knowledge

As described above, after analysing diverse domain-knowledge resources from the ER and medical domains, we conclude that domain-knowledge resources can be decomposed along three different dimensions: (i) specificity, (ii) linguistic structure, and (iii) type of knowledge resources.

At this point, it is necessary to highlight that these three dimensions are not the only possible ones in which domain-knowledge resources can be considered. For example, other possible cases might be: *language* (e.g. English, Spanish, French...), *type of data* (e.g. structure or unstructured) or *nature* (e.g. linguistic or ontological).

The combination of each dimension and its levels, results in a classification in which domain-knowledge resources can be sorted (see section 4.3). Considering that all dimension's levels are combined, adding new dimensions increases the granularity of the classification, making it more complex. This is a very expensive task that in some cases is not worthwhile. For example, in our research we have only used English resources, so adding the *language* dimension is a work that does not add any value in practise. However, if we consider multilingual OM, for example in cross-border ER scenarios, the effort of adding the *language* dimension is more than justified.

In our research, we focus on matching ontology labels from two different classifications of diseases written in English. Thus, taking into account that the number of considered dimensions should be a trade-off of effort and usefulness, we have decided to use the three dimensions mentioned above: *specificity, linguistic structure* and *type of knowledge*.

The classification produced by combining these three dimensions (see Section 4.3) has a 100% coverage classifying any domain-knowledge resource. In addition, each dimension and their levels are independent, forming disjoint sets. Therefore, each domain-knowledge resource could only fit in one part of the classification. These two characteristics indicate that the classification is complete and robust.

4.2.1 Specificity

This dimension focusses on the degree of specialisation. Here, we can find different knowledge representations regarding the level of specialisation required by the end-users. In this dimension we can identify three levels:

4.2.1.1 General

At this level is represented basic knowledge of every domain, so it contains *general/common knowledge*. For this reason, even though there are some representations of knowledge from the different domains, such as *sports, architecture, nature, or medicine*, its degree of specialisation is very low. For example, it may represent *disease* or *flu*, but not *parainfluenza virus pneumonia*. Thus, this knowledge can be used as a basis on top of which specialised knowledge can be plugged in order to generate more specific knowledge resources. An example of a general knowledge resource is WN.

4.2.1.2 Area of expertise

This level has a higher degree of specialisation than the *general level*. The knowledge represented is specific to a particular area or field (i.e. medical, architecture, ER,...). Examples of this knowledge is the content of ICD-10 and SNOMED-CT, which represent knowledge from the medical area. For example, "*Trichotillomania*" or "*Fetishism*".

4.2.1.3 Applicative

These resources have the highest degree of specialisation among the levels of specificity. The domain-knowledge at this level are instantiations of areas of expertise, so it contains the knowledge at a particular locality. For example, the knowledge that is specific to a particular hospital. Thus, members or agencies of a particular area of expertise adapt and personalise the knowledge refining it for their purposes. An example of applicative knowledge is the definition of "*emergency*" that is included in the UKCP lexicon[2] which was developed exclusively by UK agencies. Thus, it defines "*emergency*" as *an event or situation which threatens serious damage to human welfare in a place in the UK, the environment of a place in the UK, or the security of the UK or of a place in the UK.*

Interaction of Knowledge from Different Levels of Specificity

Usually, we find knowledge from different levels of specificity interacting. Indeed, this is essential in daily life, in order to properly represent knowledge and facilitate its understanding. An example of this interaction might be found in medical classifications. These resources take advantage of *general*, *area of expertise* and *applicative* domain knowledge to define the descriptions of diseases.

The main reason to address the levels of specificity lies in the ability to compose different knowledge resources and being aware of their applicability. For example, a reference glossary of medical terms, can be used in all or most medical tasks, while an application-level glossary, such as the NHS Glossary of terms of Sheffield [63], or the NHS Glossary of terms of South West London [33], are not expected to be applicable outside of their application hospitals.

The combination of levels of specificity might produce cases of ambiguity between knowledge represented at different levels. In these cases, particularly when there is a contradiction, the more specific knowledge is preferred. That is, *area of expertise* prevails over *general*, and *applicative* prevails over *area of expertise*.

Apart from the vertical heterogeneity of knowledge represented in the different levels of specificity, there might be horizontal cases of ambiguity produced at each level. In these cases the ambiguity is caused by knowledge defined differently, for example in different *areas of expertise* such as *weather*, *geography* or *architecture*, among others. Here we disambiguate by considering the knowledge represented in the *area of expertise* in which we are working (i.e. If we are matching knowledge from geographical organisations, and the matcher has the representation of *bank* from the *geography domain*, and the *architecture domain*, the representation of *bank* by the *geography domain* should prevail).

Similarly, when the ambiguity is caused by knowledge defined in different *applicative* bodies (different instantiations), the disambiguation should be done

[2] https://www.gov.uk/government/publications/emergency-responder-interoperability-lexicon

by considering the knowledge represented in the body in which we are working. (i.e. Having two English versions of SNOMED-CT, one denoted as International and the other as American, if we are in the US, the American version should prevail over the International one).

Although this disambiguation process might seem trivial, it results in extreme difficult in complex domains such as *ER*, where participants come from different areas of expertise. Thus, if the *police*, the *fire brigade* and the *ambulance service* have to automatically share information, there might be cases of ambiguity that are hard to address. An example can be found between the Spanish police and the Spanish fire brigade. Both agencies use the same "*alpha code*" in their lexicons, but whereas the former uses it for referring to a rest time, the latter uses it to represent a high priority emergency[3]. This scenario exemplifies a case in which both meanings of a term need to be considered in order to facilitate the communication between different agencies. For this reason, it is necessary to align both meanings with their respective terms in both lexicons.

4.2.2 Linguistic Structure

This dimension focusses on the linguistic part of domain-knowledge, particularly delving into the role of lexicon and grammar.

4.2.2.1 Lexicon

The lexicon is the set of all the words of a domain-knowledge language. In this context, lexicons are also called *domain terminologies.* Thus, they contain lexical terms to represent the knowledge of a specific domain.

Lexical-terms, usually encode lexical semantics, for example in lexico-semantic databases and thesauri. Thus, lexical terms are related according to their meaning, mainly by the *is-a* relation which is the principal lexico-semantic relation. This relation produces a semantic hierarchy in which terms are organised from the most general to the most specialised. For example, *disease* and *flu* are semantically related by an *is-a* relation, where *disease* is the hypernym of *flu*, and therefore *flu* is hyponym of *disease.*

[3]Information taken during my interactions with the 112 coordination centre of Jaén

4.2.2.2 Grammar

This part of linguistic structure includes the syntax, morphology and orthography of a particular domain. Grammar plays an important role in domain awareness, because depending on the domain, there are particular grammar rules. Knowing the rules of each domain helps matchers to find mappings during the matching process of domain ontologies.

Syntax

Syntax is affected by domain-knowledge because depending on the domain, words have one particular order or another. For example, in the medical domain, it is common to name disorders both using a single term, or by using an expression and the word *disorder*. Examples are: *Voyeurism* or *Voyeuristic disorder*, *Schizophrenia* or *Schizophreniform disorder*.

It is crucial to identify the grammar particularities of each domain because with this information we can generate the set of grammar rules for any domain.

Morphology

Morphology is affected by domain-knowledge because we have identified that there are prefixes and suffixes that are particular to specific domains. One example might be the process of converting adjectives into nouns in the medical domain. In this domain it is common to add the suffix *-ism*. Examples are: hyperthyroid (adj) - hyperthyroid*ism* (noun), dysthyroid (adj) - dysthyroid*ism*, or schizoid (adj) - schizoid*ism* (noun), among others.

Orthography

Domain-knowledge also contains particularities with respect to orthography, mainly by using notational conventions that differ from one domain to another. For this reason, it is essential to identify such conventions to enrich matchers with them. Thus, matchers can take advantage of them and improve the performance of the matching process avoiding ambiguity and recognising more mappings.

One example is the use of parentheses and square brackets. Depending on the case, they are used for clarification or redundancy. For example, we can find the following descriptions in DSM-5:

1. Trichotillomania (hair-pulling disorder).
2. Intellectual disability (intellectual developmental disorder) Mild.
3. No Diagnosis or Condition on Axis I / No Diagnosis on Axis II [DSM-IV].

Whereas in the first case, parentheses are used for clarification, in the second case the content within parentheses is redundant. The third case, also clarifies, indicating that this description belongs to version four of DSM-5.

Other examples are the non-standard use of uppercase/lowercase and the use of Roman numeral. For example, *Bipolar I disorder*, or *No Diagnosis or Condition on Axis I/ No Diagnosis on Axis II.* Thus, if we convert all labels in lowercase or in uppercase, there are cases in which we might lose the Roman numerals, and so, lose meaning.

4.2.3 Type of Knowledge Resource

This dimension focuses on the types of resources considering the way in which they have been generated. This is important because depending on their nature we can apply them to different approaches. In our research, we differentiate between *symbolic* and *statistical* knowledge resources.

4.2.3.1 Symbolic

These resources are created by experts of the specific fields. Generating this resources is a complex process which involves the participation of different experts who have to agree on the way in which the knowledge is represented. Besides the complexity of the process, these resources are also difficult to be maintained because knowledge evolves, so terms become obsolete and new terms are needed to be included. Moreover, adding a new entry might imply restructuring the whole resource, so it is necessary to decide what is the best way to carry out updates. An example of this complexity can be found in the UKCP lexicon where there is a record of the version in which each entry was included and additional notes to clarify its use. Examples are:

- *Access Overload Control*
    - *Acronym:* ACCOLC
    - *Included/ Revised version:* 2.0

  - *Definition:* Replaced by Mobile Telephone Privileged Access Scheme (MTPAS).
  - *Notes on definition:* This is a historical term for a scheme that no longer operates in its current form.

- *Catastrophic emergency*
  - *Acronym:*
  - *Included/ Revised version:* 2.0
  - *Definition:* An emergency which has an exceptionally high and potentially widespread impact and requires immediate central government direction and support.
  - *Notes on definition:* Defined by Cabinet Office (2010) Central Government Arrangements for Responding to an Emergency.

Regarding their degree of expressivity, symbolic resources can be differentiated from the lowest degree to the highest one:

- *Lexicons.* Repository of lexical entries, in which each entry may have some information such as definition, version of inclusion or provenance among others.
- *Thesauri.* Set of words that are structured by *synonymy* relationship.
- *WordNets.* Structured list of words organised by meaning. These resources represent word senses that are linked by semantic relationships such as *is_a* or *part_of*.
- *Domain Terminologies.* Sets of lexical-terms that belong to a particular domain. These terms are related by semantic relations.
- *Domain Ontologies.* Ontologies that consider concepts relevant to a particular domain.

4.2.3.2 Statistical

These resources are created by using statistical models which analyse huge amounts of documents of a specific domain. This is an unsupervised process that has the aim of discovering hidden relations between words, such as latent semantics [36].

Currently, the most popular statistical methodology for generating resources is word embeddings [82]. In this approach, a word is represented as a point in a multidimensional semantic space. Thus, two words are semantically close if their vectors (embeddings) are close. For example, the vector of *schizophrenia* will be closer to the vector of *brain* than to the vector of *ankle*.

Word embeddings models can be learned automatically from text. To do so, vectors are generally based on the co-occurrence matrix which represents how often words co-occur. Mainly, we can distinguish between two methods: (i) term-document matrix, and (ii) word-word matrix.

The former represents all the words as rows and the documents as columns, being the dimension of the matrix the number of documents. So, the idea is to count the number of occurrences of each word in each document.

The latter is a square matrix in which all the words are in rows and columns. This matrix counts the frequency in which each word appear close to another word in the text or texts. To do so, a window of x number of words is defined and if two words appear in the same window, the frequency of occurrence is updated by adding one occurrence.

In order to take advantage of this kind of resources in the matching process, it is necessary to train models with documents of the domain that we want to use. Doing that, these models may discover latent semantics which might not be considered in symbolic resources, because experts may not realise that these semantic relations are taking place. So, it is essential to combine both types of knowledge resources.

4.3 Combination of the Different Dimensions of Domain-Knowledge

The three dimensions of domain-knowledge described above, usually appear together in resources and methods. Below, there are described all the possible combinations (see table 4.1):

Specificity	Linguistic Structure	Type of Knowledge Resource	Resource/Method
General	Lexical	Symbolic	General Lexical Databases
General	Lexical	Statistical	General Word Embeddings
General	Grammatical	Symbolic	General Grammatical Rules
General	Grammatical	Statistical	General Machine Learning Models
Area of Expertise	Lexical	Symbolic	Domain Terminologies
Area of Expertise	Lexical	Statistical	Domain Word Embeddings
Area of Expertise	Grammatical	Symbolic	Domain-specific Grammatical Rules
Area of Expertise	Grammatical	Statistical	Domain-specific Machine Learning Models
Applicative	Lexical	Symbolic	Local Lexicons
Applicative	Lexical	Statistical	Local Word Embeddings
Applicative	Grammatical	Symbolic	Application-specific Grammatical Rules
Applicative	Grammatical	Statistical	Application-specific Machine Learning Models

Table 4.1: Resources and methods with respect to the three domain-knowledge dimensions.

General + Lexical + Symbolic

General domain-knowledge is represented by lexical terms in a resource that is manually generated by experts. Examples of these resources are general lexical databases, such as WN.

General + Lexical + Statistical

The lexical terms used to represent general knowledge are discovered automatically by applying statistical models. Resources which combine these dimensions are general word embeddings resources such as word vectors trained on Wikipedia[4].

General + Grammatical + Symbolic

The grammar used in general domain-knowledge and constructed by experts such as linguisticians. The resources that contain this combination are reference books of language grammatical rules, such as "*The Cambridge Grammar of the English Language*" [75].

General + Grammatical + Statistical

General grammar that is discovered applying statistics models. Examples of these are general machine learning models such as POS taggers, tokenisers, parsers...

[4]https://www.wikipedia.org

Area of Expertise + Lexical + Symbolic

These resources contain the knowledge of a specific area of expertise, represented by lexical terms and constructed by experts in the fields. Examples of these resources are domain terminologies such as MeSH.

Area of Expertise + Lexical + Statistical

The knowledge of a specific area of expertise is represented by lexical terms and the resource is generated by applying statistical models. Examples are domain word embeddings that have been trained with documents from the particular area of expertise, such as the PubMed[5] corpora that contains medical papers [28].

Area of Expertise + Grammatical + Symbolic

Experts generate a set of grammatical rules that are specific to a particular area of expertise. Examples are domain-specific grammatical rules, such as those contained in SPECIALIST, which contains grammar rules from the medical domain.

Area of Expertise + Grammatical + Statistical

The grammar of a particular area of expertise is identified statistically. Examples of this combination are domain-specific machine learning models, such as those trained with the PubMed corpora.

Applicative + Lexical + Symbolic

Lexical terms that are designated by experts to represent application knowledge. Examples are local lexicons, such as the UKCP which contains knowledge from the ER domain, but particular to the UK.

Applicative + Lexical + Statistical

Application knowledge represented by lexical terms is identified by applying statistical models. In this case, the resources are local word embeddings, such as those generated by training with documents from UK ER agencies.

[5] https://www.ncbi.nlm.nih.gov/pubmed/

Applicative + Grammatical + Symbolic

Application grammar generated by experts. Examples are rules for a resource or application-specific syntax and orthography, such as specific word order or specific use of punctuation.

Applicative + Grammatical + Statistical

Grammar discovered statistically by using application documents. An example might by the machine learning models that train with documents from the UK ER agencies.

4.4 Leveraging Domain-Knowledge Dimensions for Matching

Once the domain-knowledge dimensions have been introduced and their possible combinations have been described, we have defined a solution that aims at taking advantage of these dimensions to improve matching processes. Specifically, we propose generating extensions which include resources that combine the three domain-knowledge dimensions, as described above, and plugging these extensions into a matcher's KB. Figure 4.2 depicts the general scheme of our solution[6]. Each component is detailed as follows:

- *Extensions.* They contain resources generated both, symbolically and statistically, representing knowledge from the three levels of specificity and the two kinds of linguistic structure. In our research, we concentrate on generating extensions with knowledge from the *area of expertise* or *applicative* specificity levels, because these will represent specific knowledge necessary to carry out our experiments which contains domain-specific terminologies.

[6]The scheme contains three different kinds of extensions *symbolic*, *statistical* and *grammar*. This distinction may be inconsistent with respect to Section 4.3, where *grammar* is also included in *symbolic* and *statistical* resources, however this has been done on purpose to highlight the role of grammar in domain-knowledge. Thus, regarding the language structure dimension, the *symbolic* and the *statistical* extensions only contains *lexicon* knowledge.

- *Knowledge integration tool.* This is an essential component in the solution because it is the tool that allows the integration of domain-knowledge into matchers' KB. In this case, the tool should contain a general KB on top of which domain-knowledge can be plugged. The main reason for doing this, is because in the OM process it is not unusual that ontologies contain general knowledge in their representations, so that, a KB that exclusively includes domain-specific knowledge may not have represented the necessary general knowledge to discover the mappings.

- *Matchers.* These are the matchers in which the domain-knowledge extensions will be integrated. The main requirement is that they use KB to carry out the matching process.

After describing the aim and the components of our solution, below there are described several approaches, that we have defined and implemented, to integrate domain-knowledge into matchers' KB.

4.4.1 Integrating Lexical Resources

One of the problems that we have detected during our research is that when matching domain ontologies, matchers do not discover mappings because their KBs do not have the specific knowledge that is necessary to identify them. In this approach, we propose enriching the KB of the matchers with domain-specific lexical terms. To do so, we have used lexical symbolic resources from the ER and the medical domains.

In particular, we have carried out two ways in which these resources can be integrated in matchers' KB.

4.4.1.1 Rough Integration

The main idea of this type of integration is to plug a complete or partial knowledge representation of a domain-specific resource into the matcher's KB. To do so, data have to be structured, so lexical terms have to be organised in a hierarchy by an *is-a* relation. Thus, it is possible to extract subtrees of the whole resource.

The process of plugging the tree/subtree has the following steps:

1. *Selection of the subtree of lexical terms.* The root of the subtree of lexical terms is identified as well as all its descendant nodes that will be included

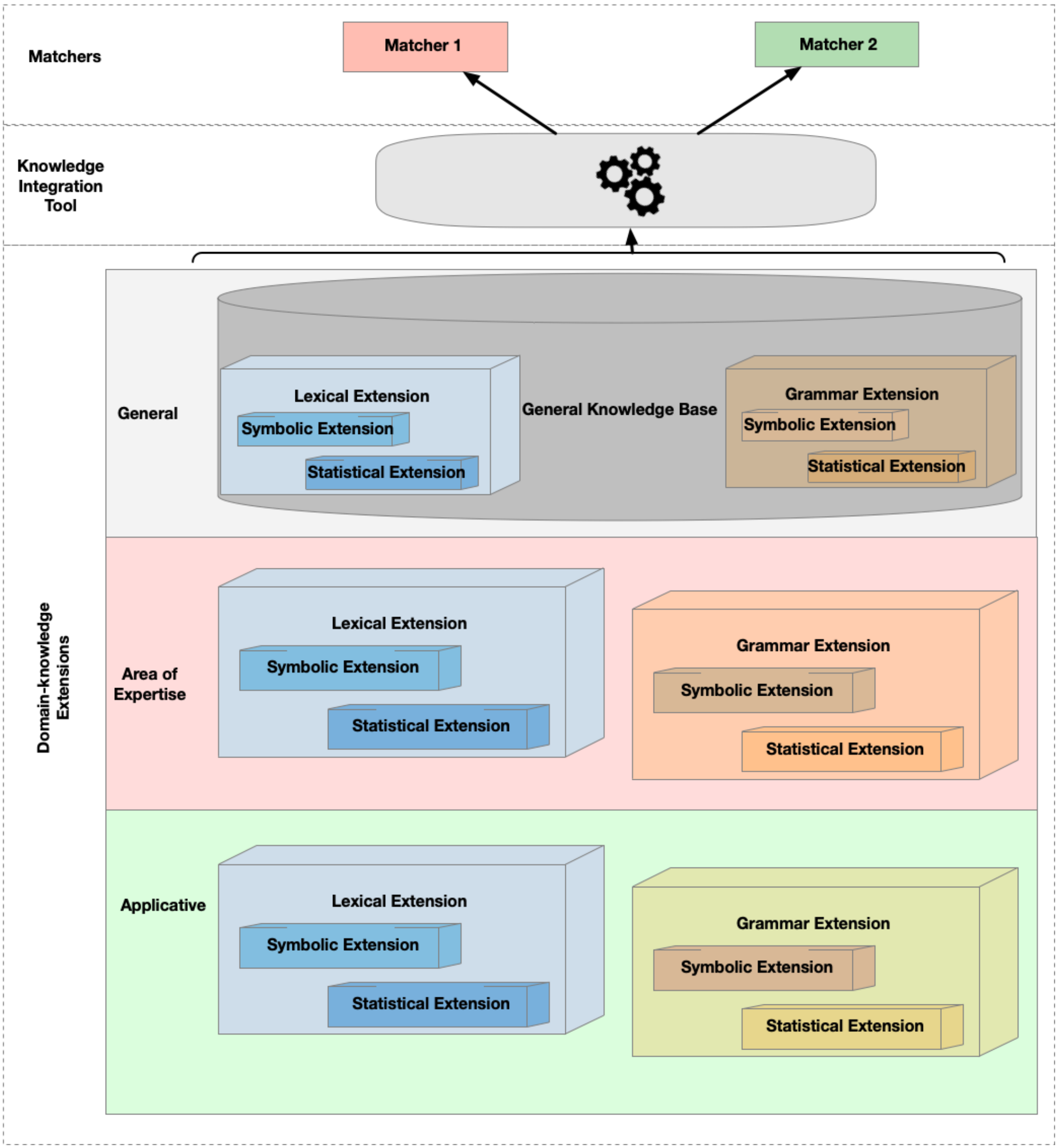

Figure 4.2: Architecture of the solution

in the extension of the KB.

2. *Annotation of relationship between terms.* For each node it is necessary to specify its parent node and its child nodes. Thus, all the lexical terms of the subtree are linked by the *is-a* relation.

3. *Representation in KB format.* Once we have all the lexical terms of the subtree linked between them, it is necessary to represent the lexical terms and the lexical semantics as they appear in the KB.

4. *Integration of the subtree into matcher's KB.* The way to link the subtree

with the matcher's KB is by linking the root with the KB by an *is-a* relation. To do so, it is necessary to identify in the KB the hypernym of the root. Because the subtree is a specialisation of the KB, it must be a lexical term which is a hypernym of the root.

4.4.1.2 Fine Integration

This type of integration aims at plugging in all kinds of data either structured or unstructured. In this case, each lexical term is analysed in order to be integrated in the best suitable place of the matcher's KB. This is a manual process which has the following steps[7]:

1. *Selection of terms.* First of all, it is necessary to select the terms to formalise. These term should be specific to the particular domain.

2. *Checking that the new terms are not represented in the matchers' KB.* Once the terms to formalise are selected, the following step involves going through every single term to verify whether it is currently represented in the KB or not. This point is crucial because if we create new concepts or senses that already exist in the KB we are increasing polysemy and redundancy, and in some cases violating the KB integrity. For this reason, we need strong arguments for creating new entries in the KB. This step has two sub-steps:

   (a) *Identifying term's hypernyms.* Analysing the definition of the given term (meaning), we can identify the hypernym of the term. This identification is important to understand the meaning of the term and how can it be integrated into the KB.

   (b) *Linking the term to existing terms in the KB.* At this point we have to seek our term in the KB and check whether it is included or not, and if it appears, it is necessary to verify whether the term's meaning in the KB is the same as our term's definition. The decisions to make, depending on each case, are described as follows:

      i. *The term is not in the KB.* After looking for the term in the BK the search did not return any result, so our term's label does not exist in the KB. At this point, two cases are possible:

[7] Due to this is a manual process, this kind of integration is only suitable when actors have the terminologies to be matched in advance.

- *The concept is currently represented in the KB*, so it is necessary to link the new term to this concept.
- *The concept is not represented in the KB*, therefore the new term is added to the KB creating an associated concept.

ii. *The term is currently in the KB.* The search of the term in the KB has retrieved one or more senses, so we have to check if any sense has the same meaning of our term. The new verification produces two new scenarios:

- *The term has the same meaning of a sense.* Because the term is currently in the KB with the same definition as our term, so that we do not have to do anything.
- *No sense has the same meaning as the term.* In this case we have to double-check if the concept is represented in the KB or not. This is the same case as the scenario described before in "2.(b).i"(*The term is not in the KB*), so we have to follow the mentioned guidelines and act depending on the case. A new sense of the word has to be added independently whether it is necessary to create a new concept or not.

3. *Double-checking term's relations.* Once all new terms are added into the KB, in this step we have to double-check that all semantic relations have been taken into account. Essentially, in the previous step we identified the semantic relations between each new term and the terms included in the KB. Nonetheless, it is also necessary to specify the relations of the new terms that are interrelated. For example, in the UKCP lexicon there are terms that include in their definitions other terms defined in the lexicon. Therefore, we have to go again through all new terms in order to ensure that all of these relations are considered.

4. *Representation in the KB.* Once we have the terms that we are going to formalise as well as their relations, the next step consists of representing these terms in the KB format. Thus, the output of this step is a file compatible with the KB.

5. *Integration of these terminology into matcher's KB.* In this step, the file which contains the specific terms is plugged in the KB.

4.4.1.3 Comparison between Rough and Fine Integrations

The suitability of each integration process depends on the nature of the resource to be integrated as well as on the needs of the organisation that will use the resource. For example, in scenarios such as ER where time is vital, it is not possible to carry out an integration process that requires much time. However, if the ER agencies are enriching their systems in advance, as a preparation task prior to an ER scenario where time is not indispensable, time-consuming integration processes may be adequate. Below, we compare both, rough and fine integration methodologies, considering the following features:

- *Kind of data.* If the resource has structured data (e.g. a taxonomy), it can be integrated using both methodologies, however, if the resource contains unstructured data (e.g. a glossary) it can only be integrated by following the fine integration method.

- *Integration process.* The rough integration process is semi-automatic, because it only needs to know the subtree to be integrated and the hypernym of the subtree's root node. However, the fine integration process requires to find all semantic relations that a new entry has and will produce on the KB after its integration. Therefore, this is a supervised process, so that, it has to be done manually.

- *Complexity.* In terms of effort and time/resource costs, on the one hand, the rough integration has low complexity, as it is necessary to only focus on the root node of the subtree to be integrated, because all the semantics within the subtree are implicitly integrated. On the other hand, fine integration demands more effort and time, as normally each new entry will not be a hyponym of a common hypernym, and so, they will be located in different places of the matcher's KB as each of entry will have a different hypernym. Moreover, in the fine integration it is also necessary to consider hyponyms of the new entry that may be included in the KB.

- *Adaptability.* Rough integration has a lower adaptability, as it is restricted to a defined resource whose semantics are inherited. However, fine integration can be adapted to the needs of the final user, as each entry is located in the KB following the criteria of the KR engineer.

- *Updates.* The update and the evolution of resources vary in the rough and in the fine integrations. The former has a static update because it only can update the KB when the resource to be integrated has a new release (e.g. a new version of MeSH). However, the latter can be updated both, when the resource has a new release or as soon as a new entry is defined to represent new knowledge (e.g. the representation of a new virus).

Table 4.2 summarises the particularities of each kind of integration.

Feature \ Method	Rough Integration	Fine Integration
Kind of data	Structured	Structured and unstructured
Integration process	Semi-automatic	Manual
Complexity	Low	High
Adaptability	Low	High
Updates	Static	Static and dynamics

Table 4.2: Comparison between Rough and Fine integration methods.

4.4.2 Integrating Grammar

This approach aims at incorporating domain-specific grammar into the matcher. To do so, it is necessary to clearly identify the set of rules that are specific to each particular domain. Below, we describe three approaches for integrating grammar in matchers' KB.

4.4.2.1 Morphological Expansion

One of the constraints that most matchers have is the problem for managing derivations of lexical terms. That is, usually they represent a lexical term and its derivations as two different concepts, so it is not possible to discover a mapping between them. In addition, this problem increases the KB with redundant knowledge.

In this approach, it is proposed to integrate into the matcher's KB morphological derivations of domain knowledge. Thus, we identified several rules to carry out morphological derivation [90] from one semantic category

(Part-of-speech (pos)) to another, by adding or subtracting particular suffixes. An example may be the following rule:

$$\textit{suffix1}\$(\text{pos1}) \rightarrow \textit{suffix2}\$(\text{pos2})$$

where the first term, formed by *lexeme+suffix1*, has pos1, whereas if *suffix2* is added to the *lexeme* we will have the second term that has pos2. In this case, both terms are morphological derivations, so if the KB has represented one of them, we should include the other term as a *derivationally related form* rather than as a new concept.

An example might be the medical term *adenohypophysis* which is a noun, and its derivation as an adjective is *adenohypophyseal*. In this case, it is followed this derivation rule:

$$physeal\$|adj|physis\$|noun$$

Therefore, having the lexeme *adenohypo*, we can have a term with an adjective pos by adding the suffix *physeal* (*adenohypophyseal*), whereas adding the suffix *physis* to the lexeme, the new term will have a noun pos (*adenohypophysis*).

Regarding the OM process, the incorporation of derivationally related words expressed in domain-specific grammar rules, will help discover new mappings that currently are not possible to be discovered. For example, when suffixes are longer than lexemes because in these cases, string similarity measures will compute a low degree of similarity and so, discard the mapping.

4.4.2.2 Syntactic Optimisation

In Section 4.2.2.2 we have seen how there are no syntactic standards or conventions used across domains. That means that each domain has its particularities for using syntax and orthography. This may also differ between the different levels of specificity, and so, the syntactic optimisation may vary depending on the domain.

In the medical domain, we have detected that the use of commas, parentheses and square brackets, directly affect the results of the matching process, so that, we propose removing commas, as well as the content within parentheses and square brackets, as a first approximation of syntactic optimisation.

Regarding parentheses and square brackets, their use is not standardised, so some times they are used to specify the version of a description and other times to define or clarify the proper description. That means that the matcher cannot take advantage of this content. In fact, currently this content adds complexity to the matching process, mainly affecting string similarity measures and the complexity of logical formulae converted from the labels.

As regards to commas, there is neither a clear use of them because there are cases in which they are used, for example, when listing different elements (e.g. different symptoms) or to add new information (e.g. *neurocognitive disorder, with behavioral disturbance*). In this case, the use of commas directly affects matchers such as S-Match because when converting labels to formulae, *commas* are used as *disjunctive connective*, what means that the matcher will output as correct mapping, for example, all medical descriptions which contains "*commas + expression*".

For this reason, carrying out the proposed syntactic optimisation reduces drastically false positives when matching domain-knowledge.

In the future we are going to explore other kinds of syntactic optimisation less aggressive than removing all parenthesis and square brackets content. One option is identifying and categorising the content within parentheses and square brackets may help the matcher to discover new mappings. For example, detecting whether the content is a synonym of the rest of the label (in that case, we can remove it). Otherwise, we can use it for matching purposes. However, this is something that is not trivial because there might be cases in which the content may contradict parts of the labels (e.g. a label that includes "*adult (paediatric)*").

4.4.2.3 Postscript Optimisation

During our analysis in domain-knowledge resource that there are cases in which several postscripts are recurrently used in descriptions. For example, in medical classifications, there are entries that add the postscript *unspecified* as a way of classifying kinds of diseases that are not precisely defined in the medical classification. An example, might be the representation of obesity in ICD-10 as we can see in Figure 4.3.

In this case, if the doctor is not sure of the kind of obesity that a patient has, he/she will choose "*Obesity, unspecified*".

The main problem with this kind of postscript is that it is metadata that does not add valuable information when the resource also is organised in a taxonomy that has *is-a* relationships between entries. Moreover, on the one hand,

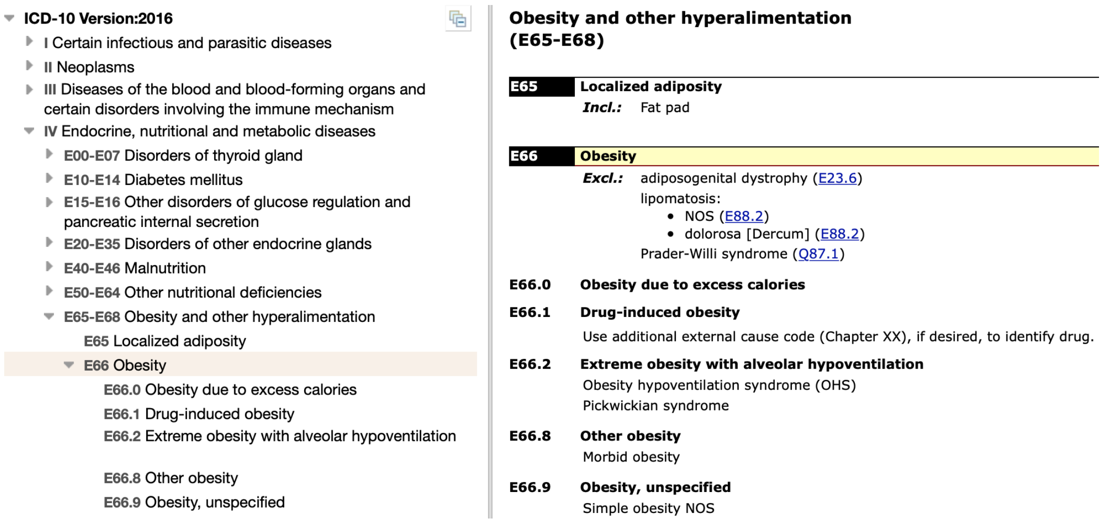

Figure 4.3: Representation of *obesity* in ICD-10

postscripts confuse matchers because if they use string similarity measures, the longer the postscript is the more string similarity it will have with other entries that have the same postscript.

For this reason, we have detected and removed postscripts within the entries of the input ontologies in a step prior to the matching process.

4.5 Summary

In this chapter we introduced our theoretical contribution. After having analysed different domain-knowledge resources we have identified and described three dimensions of domain-knowledge: *specificity, linguistic structure* and *type of knowledge resource.* Then, the variety of resources generated by the combination of these three dimensions have been detailed. After that, we described our solution to take advantage of domain-knowledge dimensions in the matching processes, defining different approaches to integrate symbolic resources and grammar into matchers' KB.

Chapter 5

Implementation

In this chapter, the implementation of the approaches presented in the theory (see Chapter 4) is described. Firstly, we have instantiated the architecture of the solution (see Section 4.4), detailing the tools and resources that have been used in each component. After that, there are detailed the extensions developed during our research, explaining how they have been generated. Concretely, we have developed one ER extension, and two medical extensions (one containing medical lexicon and another with medical grammar).

5.1 Overview

In Section 4.4, it is described the scheme of our solution, highlighting its different components and their roles in the enrichment of matchers with domain-knowledge. Below, we specify the tools and resources used for each component, justifying their selection:

- *Extensions.* We have generated three different extensions: an ER lexicon extension (see Section 5.2) which contains domain-specific terms from the UKCP lexicon; a medical lexicon extension (see Section 5.3) that includes knowledge from MeSH and SPECIALIST; and a medical grammar extension (see Section 5.4), which contains rules to expand matchers' KB morphologically and to carry out syntactic and postscripts optimisation prior to the matching process. The main reason for developing these extensions stems from the domains in which we evaluate our approaches, which firstly was ER, but due to the lack of available and accessible resources to carry out our experiments, we finally opted for the medical domain, where we used ICD-10 and DSM-5. In addition, with these three extensions, we cover most of the combinations between the three domain-knowledge dimensions described in Section 4.3, so most kinds of domain-knowledge are included.

- *Knowledge integration tool.* We have used Diversicon to plug the extensions into matchers' KB. The main reason for using this tool is because there was no other tool that fulfilled our needs and so, we decided to construct Diversicon for this particular purpose in a project in which we collaborated with people from the University of Trento[1]. Diversicon includes different tools such as *diverCli* (Diversicon client executed in the terminal), which facilitated the integration process, as well as an API to allow the easy integration with other tools. Moreover, it contains, by default, WN 3.1 as a general KB on top of which domain-knowledge can be added.

- *Matchers.* In this case, the matchers used have been S-Match and LogMap. Both matchers use BK to perform the matching, but they carry out this process differently, so we consider that it is interesting to study how this affects each matching process. Both matchers are connected with Diversicon by an adapter that was developed for each matcher.

Figure 5.1 depicts the solution scheme with the tools and resources that have been used. As we can see, domain-knowledge extensions play an essential role in our system, because the matchers will enhance their performance depending on how enriching the extensions are.

Below, we detail the extensions that we have developed during our research, highlighting the implementation of the approaches used in each case[2].

[1] Diversicon was completely developed by our partners at the University of Trento. We have contributed to the whole framework with the domain-knowledge extensions that are available in its repository and with the integration of Diversicon with LogMap.

[2] All the code used to develop the extensions and to execute the experiments is available online. http://github.com/s1580097/code

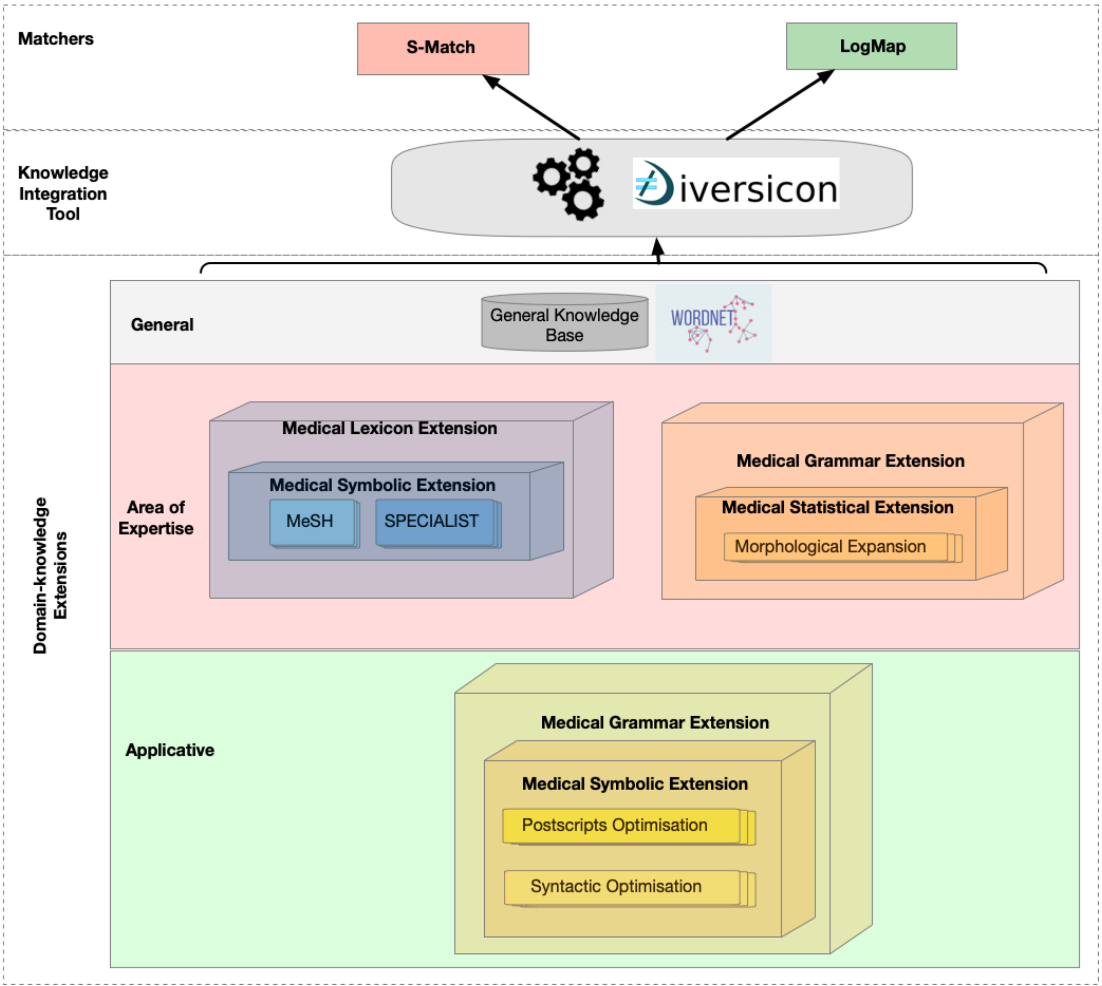

Figure 5.1: Implementation of the general schema

5.2 Emergency Response Symbolic Extension

The aim of this extension is to represent symbolic knowledge from the ER domain. To do so, we were collaborating with the Resilience Department of the Scottish Government, working with the UKCP lexicon (see Section 3.3.3.1). It is necessary to recall that this resource contains knowledge from the different levels of specificity (*general*, *area of expertise*, and *applicative*), so some of these terms are exclusively restricted to the UK (e.g. "*emergency*").

After analysing the UKCP lexicon and discussing with practitioners of the Resilience Department of the Scottish Government, we selected a subset of 100 lexical entries which includes the most representative terms used by UK ER agencies. These terms were used to develop the ER extension by using the fine integration procedure described in Section 4.4.1.2. The main reason for selecting this integration methodology was the nature of data which is unstructured. Thus, integrating each lexical entry entailed the process of finding the precise synset that is its hypernym in WN, and consider its semantics relations in both within WN and between other terms of the extension (see Section 4.4.1.2). The whole process was challenging as it involved multiple interactions with ER practitioners to refine the extension until its release.

One of the major issues that we encountered is the ambiguity of terms that are currently represented in WN, but have a different meaning in the domain-specific resource, in this case, in the UKCP lexicon. An example is "*evacuation*", whose definition in WN does not cover the definition in the UKCP. In these cases, we opted for adding a new synset in WN that fulfils the domain-specific definition (see Figure 5.3). The main reason is that we consider that "*applicative*" knowledge should prevail over "*area of expertise*" knowledge and "*area of expertise*" knowledge should prevail over "*general*" knowledge, because this is the way to preserve the knowledge with highest degree of specificity. This also allows matchers to select which kind of knowledge they want to use depending on the resources that they are matching.

Figure 5.2 shows an extract of the terms included in the extension. We can see several examples of terms from the different levels of specificity: general (*"extranet", "disaster", "downstream"*), area of expertise (*"casualty receiving hospital", "command and control level"*) , and applicative ("*level 1 emergency*, "*category 1 responder*").

The extension was developed following the LMF format [47, 48] (see Section 2.2.3), which is the format in which WN is codified. After that, the extension was integrated into WN [118, 119]. It is important to remark that even though the UKCP lexicon contains natural language terms, we considered the semantic relations between the terms in the extension and the entries within WN. Figure 5.3 depicts the semantic relations between the entry *evacuation* as it is defined in the UKCP (drawn in blue), the related entries in WN (drawn in black), and other entries of the extension (drawn in blue).

In this case, considering the definition in UKCP, we identified "*withdrawal*" as a hypernym of "*evacuation*". After that, the kind of evacuations represented in WN, ("*medical evacuation*"), were linked as hyponyms of "*evacuation*". Finally, we added as hyponyms other kinds of evacuations which appear in the extension: "*large scale evacuation*", "*mass evacuation*" and "*small evacuation*".

4x4 response	Access control point	Accident	Agency	Air ambulance
Airwave	Assambly point	Assistance centre	Back-up	Blue route
Body collection point	Body holding area	Body viewing area	Bronze	Capability
Casualty	Casualty bureau	Casualty clearing officer	Casualty form	Casualty information unit
Casualty receiving hospital	Catastrophic emergency	Category 1 responder	Category 2 responder	Cold zone
Command	Command and control	Command and control level	Command protocol	Control centre
Control room	Controlled area	Coxswain	Crisis	Critical incident
Crowd doctor	Cruse	Decontamination point	Designated receiving hospital	Disaster
Downstream	Emergency	Evacuation	Evacuation assembly point	Event
Exclusion zone	Exercise	Extranet	Gold	Helpline
Holding area	Host organisation	Hot zone	Hub	Incident
Inner cordon	Large scale evacuation	Lead responder	Level 1 emergency	Level 2 emergency
Level 3 emergency	Level of emergency	Loading point	Local	Local emergency
Major accident	Major incident	Marshalling area	Mass evacuation	Medevac
Media	Media Liaison officer	Media Liaison point	Media Protocol	Medium-scale evacuation

Figure 5.2: Extract of terms included in the ER symbolic extension.

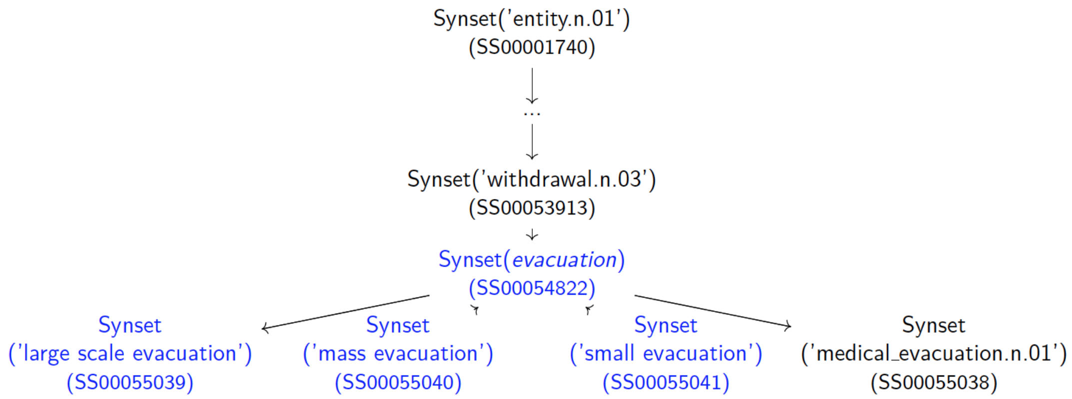

Figure 5.3: Example of semantic relations in the extension

The extension is publicly available on the Diversicon website[3].

[3] http://diversicon-kb.eu/dataset/wordnet-extension-emergency-response

5.3 Medical Lexicon Extension

This extension is composed of two extensions developed from the medical resources MeSH and SPECIALIST.

5.3.1 MeSH Extension

Because, in our experiments, we focused on matching *mental health* descriptions, this extension is a subset of the complete MeSH, only including the headings which represent *Disorders* and knowledge of *Psychiatry and Psychology*. The extension has been generated in LMF and is also available on the Diversicon website.

As detailed in Section 3.3.2.1, all elements in MeSH are sorted by hierarchies that follow either the *is_a* or *part_of* semantic relations depending on their category (e.g. the Anatomy category follows *part_of* whereas the Diseases category uses *is_a*). That means that the resource is structured in a tree shape, therefore, it is possible to extract a part of it and apply the *rough integration* described in Section 4.4.1.1. In particular, we extracted the subtrees of all kinds of diseases and appended them as hyponyms of the synset *disease* in WN. It is necessary to highlight that the categories that include the diseases are organised only following the *is_a* semantic relation, so this helps us to integrate them into WN.

Figure 5.4 shows an example of the integration of MeSH into WN. All elements of the extension, entries and relations, are drawn in blue, whereas those drawn in black are synsets and relations defined in WN.

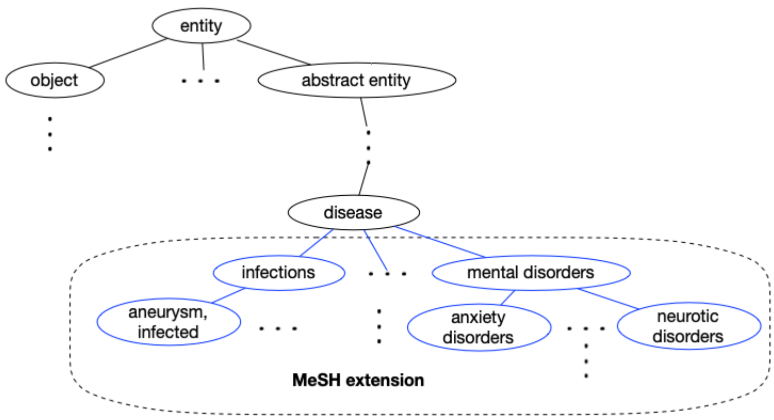

Figure 5.4: Integration of MeSH extension into WN.

5.3.2 Specialist Extension

For this extension we formalised in LMF the whole resource, which contains more than 120,000 lexical entries. These entries are related between them by the *synonymy* semantic relation. That means that they can be grouped into sets of synonyms ("*bags of words*") in a similar way to what WN does with *synsets*. The huge number of lexical entries to integrate totally ruled out the *fine integration* option, but in contrast to MeSH, in this case the resource is not sorted in a tree shape, so in principle it is not possible to use the *rough integration* option. In order to address the tree shape issue, we plugged the entries into WN by creating a subtree with 2 levels of synsets. The first level is a synset called *Specialist root*, which is linked with "*entity*" that is the most abstract synset in WN. The second level contains all the "*bag of words*" of SPECIALIST and are linked as hyponyms of the *Specialist root* (see Figure 5.5).

Figure 5.5 depicts an example of the integration of the SPECIALIST extension into WN. Each oval represents a synset, and all the lexical entries that are included in a synset are represented as elements of a set within curly brackets. All the ovals and links draws in black are synsets and relationships defined in WN, whereas those in blue are elements of the extension. We can see how, SPECIALIST includes "*stereotypy*", and "*stereotype, stereotypic movement disorder*" and "*stereotyped behaviour*" as synonyms. Therefore, "*stereotypy*" is used to denote the synset which contains its lexical entry and the ones of its synonyms .

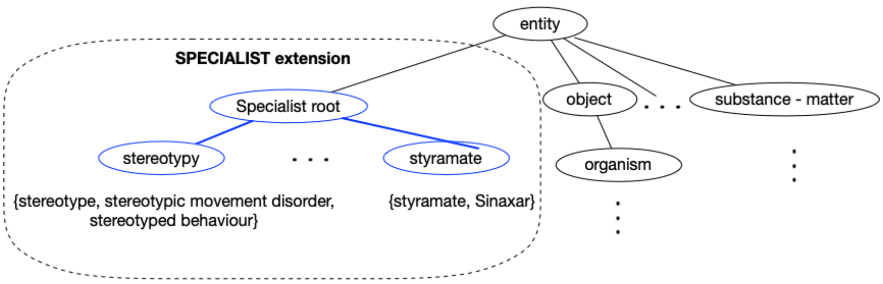

Figure 5.5: Integration of Specialist extension into WN.

Similarly, "*styramate*" includes "*styramate*", which is the chemical compound, and "*Sinaxar*", that is the name this drug commercialised by an specific brand.

5.4 Medical Grammar Extension

In this section, there are detailed the three parts which form the medical grammar extension. In particular, we highlight the decisions made and how each part was implemented.

5.4.1 Morphological Expansion

The main purpose of this extension is that matchers can take advantage of the morphological expansion of biomedical terms to discover new mappings. To do so, we have integrated several grammar rules specific to the medical domain. These rules are included in SPECIALIST and were statistically generated by the Lexical Systems Group which is the committee of organisations in charge of maintaining the SPECIALIST lexicon.

The rules define how to derive morphology from one pos to another, by adding or subtracting particular suffixes. In particular, the rules implemented are defined by the following format:

$$\textit{suffix1}\$(\text{pos1}) \rightarrow \textit{suffix2}\$(\text{pos2})$$

First, the lexical entry which has *suffix1* and whose pos is pos1 is indicated. The way to convert the lexical entry from pos1 to pos2 is by removing *suffix1* and appending *suffix2*. This produces a new lexical entry with a morphological derivation.

The $ symbol indicates the end of the suffix. There are some cases, in which the lexical entries does not have *suffix1*, so the morphological derivation is produced by appending *suffix2*.

Below are defined the 14 grammar rules included in our extension and an example of each rule. The main reason for following these rules is because they affect around 9,000 instances in total, producing more than 97% of precision within the SPECIALIST resource.

These rules have been extracted by analysing thousands of medical documents semi-automatically by the National Library of Medicine and are available within the SPECIALIST lexicon. Although most of words satisfy these rules there are some exceptions. For this reason, each grammar rule includes the total number of occurrences of lexical entries that are applicable for the rule, the number of correct instances and the number of possible exceptions of the rule. After that, the *precision* of the rule is computed. Finally, an example of each rule is included.

- iance$(noun)→ iant$(adj)
  - *Instances: 34*
  - *Occurrences: 34*
  - *Exceptions: 0*
  - *Precision: 100%*
  - *Example:* irradiance(noun) → irradiant(adj)
- iency$(noun) → ient$(adj)
  - *Instances: 55*
  - *Occurrences: 55*
  - *Exceptions: 0*
  - *Precision: 100%*
  - *Example:* immuno-deficiency(noun) → immuno-deficient(adj)
- ization$(noun) → ize$(verb)
  - *Instances: 1250*
  - *Occurrences: 1250*
  - *Exceptions: 0*
  - *Precision: 100%*
  - *Example:* tuberculization(noun) → tuberculize(verb)
- physeal$(adj) → physis$(noun)
  - *Instances: 26*
  - *Occurrences: 26*
  - *Exceptions: 0*
  - *Precision: 100%*
  - *Example:* adenohypophyseal(adj) → adenohypophysis(noun)
- sation$(noun) → ze$(verb)

  - *Instances: 1081*
  - *Occurrences: 1081*
  - *Exceptions: 0*
  - *Precision: 100%*
  - *Example:* autoimmunisation(noun) → autoimmunize(verb)

- sation$(noun) → zed$(adj)
  - *Instances: 131*
  - *Occurrences: 131*
  - *Exceptions: 0*
  - *Precision: 100%*
  - *Example:* anesthetisation(noun) → anesthetized(adj)

- se$(verb) → zation$(noun)
  - *Instances: 1118*
  - *Occurrences: 1118*
  - *Exceptions: 0*
  - *Precision: 100%*
  - *Example:* aerosolise(verb) → aerosolization(noun)

- sed$(adj) → zation$(noun)
  - *Instances: 130*
  - *Occurrences: 130*
  - *Exceptions: 0*
  - *Precision: 100%*
  - *Example:* ventricularised(adj) → ventricularization(noun)

- ability$(noun) → able$(adj)
  - *Instances: 1294*
  - *Occurrences: 1294*
  - *Exceptions: 0*

  - *Precision: 100%*
  - *Example:* non-coagulability(noun) → non-coagulable(adj)

- $(adj) → ness$(noun)
  - *Instances: 2737*
  - *Occurrences: 2737*
  - *Exceptions: 0*
  - *Precision: 100%*
  - *Example:* bloodless(adj) → bloodlessness(noun)

- de$(verb) → sion$(noun)
  - *Instances: 62*
  - *Occurrences: 62*
  - *Exceptions: 0*
  - *Precision: 100%*
  - *Example:* occlude(verb) → occlusion(noun)

- ence$(noun) → ential$(adj)
  - *Instances: 43*
  - *Occurrences: 43*
  - *Exceptions: 0*
  - *Precision: 100%*
  - *Example:* pestilence(noun) → pestilential(adj)

- ical$(adj) → y$(noun)
  - *Instances: 796*
  - *Occurrences: 788*
  - *Exceptions: 9*
  - *Precision: 98.87%*
  - *Example:* uroradiological(adj) → uroradiology(noun)

- ism$(noun) → istic$(adj)
  - *Instances: 234*
  - *Occurrences: 228*
  - *Exceptions: 6*
  - *Precision: 97.44%*
  - *Example:* fetichism(noun) → fetichistic(adj)

The percentages are calculated considering:

$$Precision = \frac{Occurrences}{Instances} \tag{5.1}$$

Notice that the precision in the last two rules is not 100%. The reason is that there are some cases in which the rules are not applicable (exceptions), producing words that do not exist in English. Thus, this fact produces false positives. Examples of these cases are:

- pollical(adj) → polly(noun)
- organical(adj) → organy(noun)
- organism(noun) → organistic(adj)

In our case, we used the file with the identified exceptions and removed them from our extension.

This extension has been generated in LMF to allow its integration into WN. However, the integration is not trivial, because there are two possible scenarios: the morphological derivation belongs to a term that is represented in WN or not. In the former case, the integration is done by linking the derivation and the term with the *derivationally related form* relation. In the latter, it is necessary to represent in WN both, the morphological derivation and the term for which it is derived. In this case, we followed the same idea of the SPECIALIST extension (see Section 5.3.2), generating "*bags of words*" and carrying out a rough integration.

5.4.2 Syntactic Optimisation

As it is described in Chapter 4, orthography conventions and standards may vary depending on the level of specificity of a particular domain. In the medical domain we have identified a non-standard use of parentheses, square brackets and commas.

Examples of the use of parentheses and square brackets might be the following:

1. Sleep terrors [night terrors]

2. Premature (early) ejaculation

3. Obstructive sleep apnea (adult) (pediatric)

In the first case, the square brackets are used to specify an equivalent expression of "*sleep terrors*". Similarly, in the second case, the content within parentheses is used as a synonym of "*premature*", so it adds redundant information. In the third case, the parentheses are used to indicate the applicability of the description, which can be used to categorise both adults and children.

Matchers cannot distinguish between these diverse uses. In fact, this heterogeneity considerably affects negatively their performance.

For example, both, S-Match and LogMap, process every bracket in the same way. The former adds the content to the label formula by using conjunction connectives. For example, the matcher translates example 3 as:

Obstructive & sleep apnea & adult & pediatric

The latter extracts the content, to compute the context of the labels. Thus, in both cases, matchers are affected as *adult* and *pediatric* are antonyms and so, belong to different contexts.

For this reason, we made the decision to remove all content within parentheses and square brackets from input medical descriptions in a preprocessing step before starting the matching process. This also implies the elimination of this content from the gold standard.

Similarly, as in the use of parentheses, commas are utilised with different purposes in medical-domain knowledge. Below there are some examples:

1. Tobacco use disorder, Mild

2. Adverse effect of unspecified antidepressants, sequela

3. Circadian rhythm sleep disorder, shift work

In example 1, the comma is used to specify the degree of the disorder, whereas in example 2 it is used to define the kind of adverse effect. Finally, in example 3 the comma is used to specify the cause of the disorder.

As we have seen before with the use of parentheses, there is not a common criterion or standard, so this adds difficulty to the matching process.

Apart from that, the use of commas has a significant impact on the results of S-Match because it translates commas into disjunctions when computing the label formula. An example of this might be *Tobacco use disorder, Mild*, whose label formula is:

$$tobacco \ \& \ use \ \& \ disorder \mid Mild$$

This also produces the following node formula:

$$(tobacco| \ Mild) \ \& \ (use \mid Mild) \ \& \ (\ disorder| \ Mild)$$

Thus, the matcher will output as mapping with this description, any other disorder that include ", Mild" in its description. For example:

$$Tobacco \ use \ disorder, \ Mild \equiv Alcohol \ use \ disorder, \ Mild$$

which is totally incorrect.

Similarly, as we previously decided with parentheses and square bracket, we have made the decision of removing the commas from the labels within input ontologies and the gold standard.

Table 5.1 shows the frequency of parentheses, square brackets and commas within DSM-5 and ICD-10. We can see how the resources differ in the way in which they represent knowledge. This shows the impact of applicative grammar in each case.

	DSM-5	ICD-10
Parentheses	96	3
Square brackets	1	5
Commas	679	198

Table 5.1: Frequency of punctuation in the datasets

The figures, correspond to the total number of occurrences within each resource and there are cases in which a description contains more than one punctuation mark (e.g. descriptions with 2 or more commas).

5.4.3 Postscripts Optimisation

In our analysis of medical resources (ICD-10 and DSM-5), we have detected that the postscripts "*, unspecified*" and "*, so stated*" recurrently appears in medical descriptions. The former is used in order to give clinicians a general category that they can use if they need to categorise a disease and it is not specifically defined. We can see an example of this in Figure 5.6. In that case, *F44.9* is defined as a general category to include any dissociative and conversion disorder that has not been specified above.

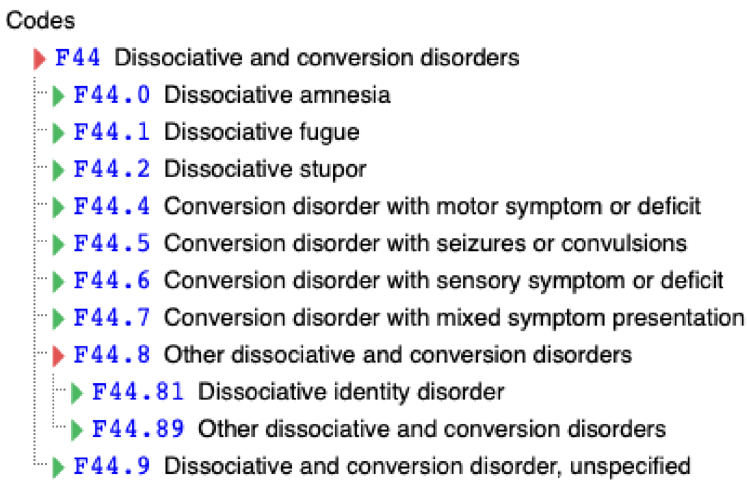

Figure 5.6: Example of postscript in ICD10

On the one hand, matchers such as S-Match, which consider subsumption relations, can identify whether one concept is more general or more specific than other. Indeed, this is the same effect that is implied in the medical domain by using the "*, unspecified*" postscript.

On the other hand, for matchers that mainly use string similarity measures, such as LogMap, postscripts increase the number of characters, adding noise that may confuse the matcher to find a correct mapping.

Table 5.1 shows the frequency of postscripts in the DSM-5 and ICD-10.

In this case, the use of "*, so stated*" is insignificant compared to "*, unspecified*", however it produces a huge number of false positives. The reason is that when

	DSM-5	ICD-10
, unspecified	8	153
, so stated	0	1

Table 5.2: Frequency of postscripts in the datasets

the matcher lemmatises "*,so state*" it outputs the lemma "*state*" and "*state*" in WN is a too abstract synset with many hyponyms (e.g. *feeling*, *skillfulness*, *condition...*). As a result, when the matcher finds any of these hyponyms in other labels, it concludes that there is a subsumption relation and so, outputs the mapping.

Considering the previous issues, we decided to remove these from the description of diseases within the input ontologies by adding them into matchers' stop words[4].

Currently, postscripts are identified in a supervised process in which we analyse false positives and false negatives of the experiments carried out in Chapter 6 looking for patterns particular to the medical domain. However, this process can be done automatically in the future by identifying these patterns within the input ontologies (see Section 7.2.3).

5.5 Summary

In this chapter, we have described the implementation of the solution proposed in Chapter 4. First of all, we have instantiated the proposed solution (see Section 4.4) with specific tools and resources. In particular, we have used S-Match and LogMap as matchers, Diversicon as a plug-in tool and three different extensions as domain-knowledge extensions (one symbolic ER extension, one lexicon medical extension and one extension with medical grammar). Each extension has been explained in detail, highlighting how it was developed and justifying the decisions that we made.

[4]These are words that are filtered out before the matching process.

Chapter 6

Evaluation

In this chapter, we evaluate our hypothesis, which was presented in Chapter 1:

"*When matching domain ontologies, matchers with DA functionality have a better performance in terms of precision and recall than those which do not have this functionality because domain knowledge helps them to disambiguate and discover mappings that otherwise could not be found, and reject mismatches that look superficially plausible.*"

The hypothesis is evaluated by an experiment, in which we match two of the most used resources from the medical domain. One is specific to mental health whereas the other is a general classification of diseases. The main purpose of executing this experiment is to prove the claims of our hypothesis in a real scenario. To do so, we compare the performance between matchers' vanilla version and the versions in which they take advantage of domain-knowledge extensions.

The chapter is organised as follows. First of all, the resources used in the evaluation are pointed out, highlighting the matchers, the KB extensions and the medical resources that have been used. Secondly, we define the metrics used for the evaluation. After that, the results of two experiments are explained and analysed. The chapter finishes with a discussion about the results and a brief summary.

6.1 Evaluation Resources

The hypothesis has been evaluated by using two different matchers whose KBs have been enriched with two domain-specific extensions: a lexicon and a grammar extension. These resources, which were explained in chapters 3 and 5, are outlined as follows:

- *Matchers.* The matchers used are S-Match [58] (see Section 3.1.2.1), and LogMap [79] (see Section 3.1.2.2). The main reason for selecting these two matchers lies in the different ways in which they perform the matching process.

- *KB medical extensions.* Matchers' KBs were enriched with two different medical extensions: a medical lexicon extension which integrates MeSH and SPECIALIST (see Section 5.3); and a medical grammar extension that has three parts: derivational morphology of medical terms, syntactic optimisation and postscript optimisation (see Section 5.4).

- *Medical resources to be matched.* Due to the medical domain is enormous, we have narrowed down our scope, only focussing on mental health disorders because this is a subdomain of special interest for professionals from different areas of expertise such as medicine and psychology. Thus, DA matching is evaluated by aligning some structures of two of the most popular medical classifications of diseases: ICD-10 (see Section 3.4.1) and DSM-5 (see Section 3.4.2), which is the reference manual for mental health disorders. To do so, we have firstly used a source dataset with all entries included in DSM-5 (743 entries) and a target dataset with all ICD-10 entries that are specified in DSM-5 as correspondences of the entries within the source dataset (591 entries). For a second experiment we have randomly selected a dataset with 200 entries extracted from DSM-5, and a target dataset with 177 descriptions included in ICD-10, which are the correspondences of the entries selected from DSM-5. The only restriction was that each description cannot contains more than 8 words to be a candidate of the source dataset.

- *Gold standard.* The results are evaluated by using as gold standard the correspondences between ICD-10 and DSM-5 published in DSM-5, where it is specified to which code in ICD-10 corresponds each description in DSM-5.

The experiments have been executed in a computer with a 3,7 GHz 6-Core Intel Core-i5 processor and 8 GB 2667 MHz DDR4 RAM memory. It is necessary to highlight that our experiments needed a powerful computer due to memory requirements, particularly when using S-Match because this matcher makes an extensive use of RAM memory where all labels and nodes formulae, and the mappings are simultaneously loaded.

6.2 Metrics

The evaluation process has been carried out considering the most relevant measures in IR [11, 95]: *precision*, *recall* and *f-measure*.

The *precision* is the proportion of positive identifications that are correct. In our case, the proportion of output mappings that are included in the gold standard.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{6.1}$$

The *recall* is the proportion of actual positives that are correctly identified. In our case, the proportion of output mappings that are in the gold standard, compared with all mappings included in the gold standard.

$$\mathrm{Recall} = \frac{True\ Positives}{True\ Positives\ +\ False\ Negatives} \tag{6.2}$$

Analysing *precision* and *recall* separately might produce a misinterpretation of the matcher's performance. For example, if a matcher only discovers one mapping, and this is correct, it will have 100% *precision* as it has zero *false positives*. However, *recall*, will be really low (assuming that there are many mappings between the input ontologies, and not only one). Similarly, a matcher that retrieves all possible combinations of the input ontologies entries will have 100% *recall*, but a low precision. For this reason, in order to avoid these misunderstandings, we use the *f-measure*, which is the standard measure, used in the literature, that combines *precision* and *recall*.

$$F\text{-}measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{6.3}$$

In these three measures, the higher the numbers are, the better the performance is.

The minimum acceptable value, will depend on the application domain. There are cases in which *precision* is more important than *recall*, and others in which it is the other way round. For example, discussing about this with practitioners of

ER agencies, they said that it should be a trade-off between the two of them. On the one hand, they cannot handle many different pieces of information because this may be counter-productive adding complexity to the decision making process. Therefore, they require specific and precise information. However, on the other hand, they have to be aware of every emergency, so they need a good *recall* in order to avoid that an emergency gets unnoticed.

In our research, we do not have any constraint for the minimum, *f-measure*, except that it should improve the baseline's one, that is, the *f-measure* of the current versions of S-Match and LogMap.

In order to avoid redundant mappings that may alter the results by considering twice a relation discovered by S-Match (*equivalence* and *subsumption*), we use S-Match minimal mappings. This S-Match configuration extracts a subset of the whole set of correspondences found between two input ontologies, containing the minimum number of mappings to represent all the properties of the initial set. Thus, none of these mappings can be dropped without loosing a property of the initial set.

6.3 Evaluation Methodology

As has been said before, S-Match and LogMap are two matchers that perform the matching process differently, producing different outputs. Whereas the former outputs the alignment indicating different semantic relations (e.g. *equivalence, subsumption, disjoint*), the latter only outputs mappings that hold an *equivalence* relationship.

Considering this diversity of possible outputs, we compute *true positives* following the convention adopted by the OAEI 2013 benchmark and conference track [62], in which correspondences only consider *equivalence* relations. That means that if a matcher computes that two entries have a similarity degree above a defined threshold, they are output as a mapping and so, it is considered that they hold an *equivalence* relation. In our case, due to the fact that S-Match outputs not only *equivalences*, but also *subsumptions*, we consider a correct mapping either when it outputs an *equivalence* or a *subsumption*, and this relation appears in the gold standard[1]. The main reason of making this decision is that no matter the relation identified by S-Match, there is a high degree of similarity between the

[1]The gold standard only contains correspondences that hold an *equivalence* relationship.

identified elements. Therefore, only considering *equivalence* mappings unfairly penalises the matcher if we compare it with other matchers that only make dychotomic decisions (mapping or not mapping). We show below the results of the experiments with S-Match regarding both cases: (i) considering only *equivalences* (see Tables 6.1 and 6.4), and (ii) considering both *equivalences* and *subsumptions* (see Tables 6.2 and 6.5).

For the first case, we extract from the output the mappings that have an *equivalent* relation and use them to compute the metrics. In the second case, we do the same, but adding also *subsumptions*.

6.4 Experiment 1.- Whole DSM-5 Dataset

The aim of the experiment is to align mental health disorders extracted from DSM-5 and ICD-10. The dataset has two structures, the first one with 743 descriptions from DSM-5 and the second one with 591 descriptions from ICD-10. These structures are aligned by using S-Match and LogMap matchers with their current KB (vanilla version), and with the two medical extensions (lexicon and grammar). The results are evaluated by comparing them with the gold standard, in which are defined 743 correspondences between both structures. Therefore, within the datasets, there are some elements from ICD-10 that have more than one correspondence from DSM-5.

6.4.1 Results

Below are tabulated the *true positives, false positives* and *false negatives* obtained after running the experiment[2] (see Tables 6.1, 6.2 and Table 6.3). Using these results, the previous metrics (*precision*, *recall* and *f-measure*) have been calculated and depicted in a heat map diagram (see Figures 6.1, 6.2, and 6.3).

[2] It necessary to highlight that the total number of correspondences is the sum of *true positives* and *false negatives* (in our case, 743). However, the number of *false positives* might be, in the worse case, $((n \times m) - n)$, being n the size of the source ontology and m the size of the target ontology. Therefore, it is possible to have a number of *false positives* bigger than the total number of correspondences.

6.4.1.1 S-Match

Only Equivalences

	Total	True Positives	False Positives	False negatives
S-Match	743	79	3	664
S-Match (Lexicon extension)	743	54	1	689
S-Match (Grammar extension)	743	103	6	640
S-Match (Both extensions)	743	100	3	643

Table 6.1: Results experiment 1 with S-Match considering only equivalences

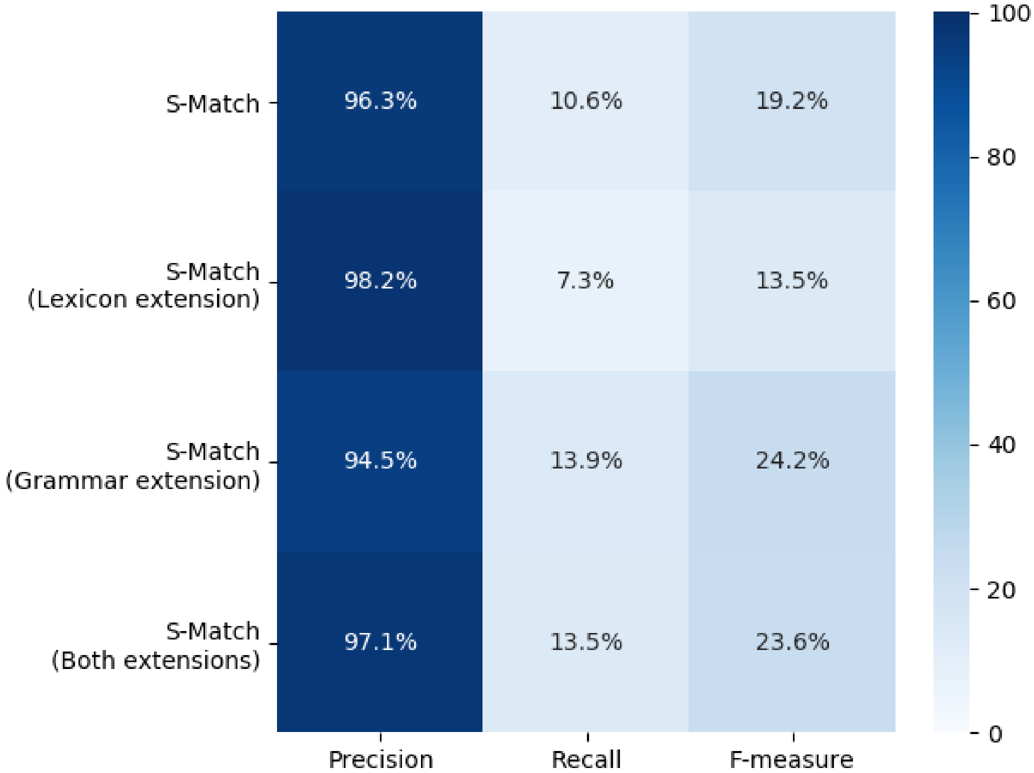

Figure 6.1: Metrics results experiment 1 with S-Match considering only equivalences

Equivalences and Subsumptions

	Total	True Positives	False Positives	False negatives
S-Match	743	315	5709	428
S-Match (Lexicon extension)	743	277	4701	466
S-Match (Grammar extension)	743	228	777	445
S-Match (Both extensions)	743	226	589	447

Table 6.2: Results experiment 1 with S-Match considering equivalences and subsumptions

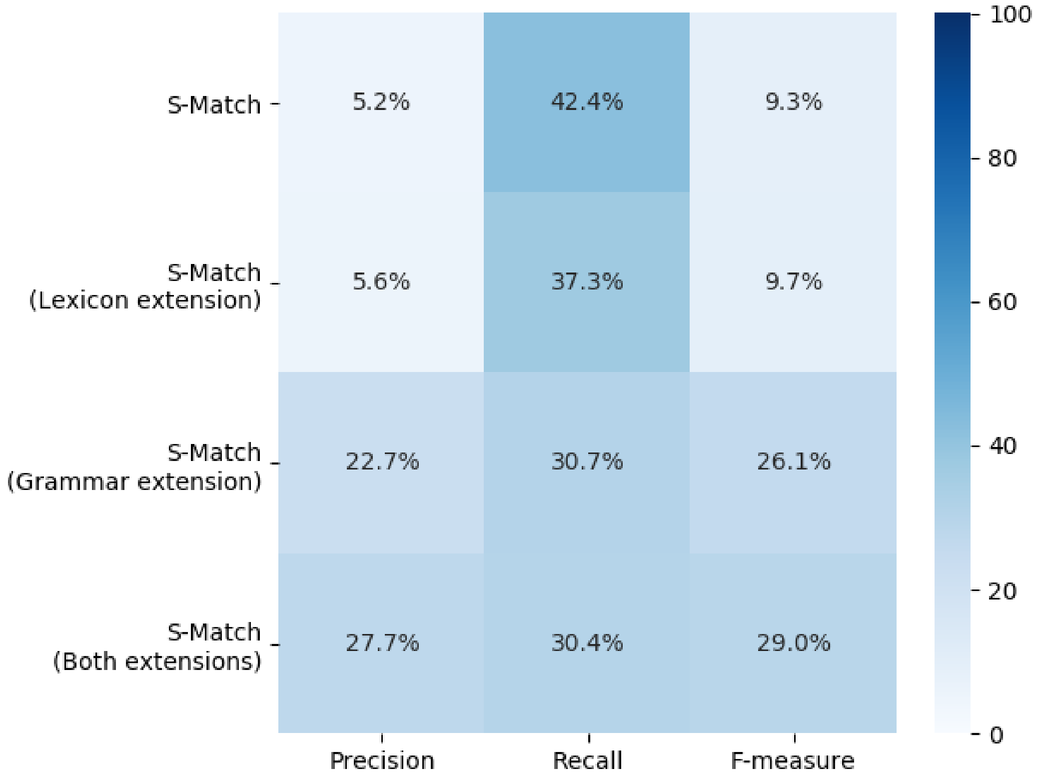

Figure 6.2: Metrics results experiment 1 with S-Match considering equivalences and subsumptions

6.4.1.2 LogMap

	Total	True Positives	False Positives	False negatives
LogMap	743	129	95	614
LogMap (Lexicon extension)	743	143	109	600
LogMap (Grammar extension)	743	143	119	600
LogMap (Both extensions)	743	141	118	602

Table 6.3: Results experiment 1 with LogMap

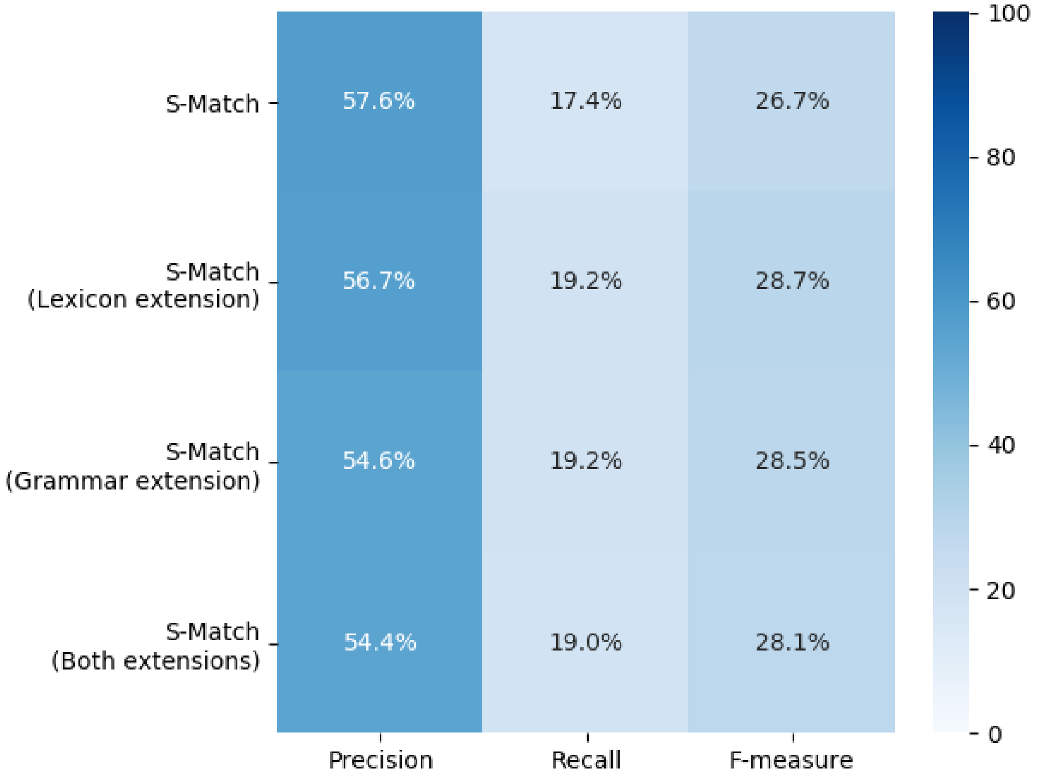

Figure 6.3: Metrics results experiment 1 with LogMap

6.5 Analysis Experiment 1

The results gathered in the previous section are analysed in detailed below. Mainly, we focus on the impact that the extensions have on the matching process and the causes that produce the obtained results.

6.5.1 S-Match

Below are analysed the results obtained after executing the experiments with S-Match. These results have been evaluated twice. Firstly, considering only equivalences as true positives and secondly, considering both equivalences and subsumptions.

6.5.1.1 Only Equivalences

Table 6.1 and Figure 6.1 show the results of the experiment executed in S-Match, considering only equivalences as true positives. We can see how the performance of S-Match slightly improves its *precision* when using mainly lexicon medical extensions. However, it obtains the best recall with the grammar medical extensions.

The most striking result from the vanilla version of S-Match (baseline, from this point on) is its low recall, which is caused by two reasons. Firstly, the matcher is only focussed on discovering "perfect mappings" between labels, so if there is any word of a source label that does not match with a target label, the matcher will discard this mapping. Secondly, the dataset contains in some cases large descriptions as labels, so this makes more complex the matching process and improbable to find perfect mappings.

In terms of *precision*, the baseline has a good performance more than 96%. Mainly, because the matcher only outputs mappings considered as equivalent, so the similarity degree between them is really high. This results in having only few false positives as we can see in Table 6.1.

As a result, considering the general performance, *precision* is penalised by a lower *recall*, around 10%, resulting a *f-measure* around 19%.

Lexicon extension

In this case, the matcher's KB is enriched with medical knowledge from SPECIALIST and MeSH. We can see how with this domain knowledge the number of false positives has reduced around 66%, improving *precision* around 2% with respect to the baseline, but *recall* has considerably decreased more than 31.5%. The main reason of this behaviour is that there are terms that appear in the extension, but do not have any other related form within the KB. Therefore, it is not possible to find a mapping. An example of mapping that is not discovered with the lexicon extension, but was discovered with the vanilla version is:

- "*Stereotypic movement disorder*" ≡ "*Stereotyped movement disorders*"

In the vanilla version, S-Match computed label formulae by extracting the lemmas contained in the label. In this case, both are the same:

$$(\textit{stereotype}\ \&\ \textit{movement}\ \&\ \textit{disorder}) \equiv (\textit{stereotype}\ \&\ \textit{movement}\ \&\ \textit{disorder})$$

However, the lexicon extension has an entry with the lemma "*stereotypic movement disorder*". In this case, the label formulae are as follows:

$$(\textit{stereotypic movement disorder}) \equiv (\textit{stereotype}\ \&\ \textit{movement}\ \&\ \textit{disorder})$$

That means that, this label formula with a compound lemma currently cannot be matched in S-Match with a label formula that represents lemmas individually. This helps to reduce false positives, but also penalises the matcher not discovering valid mappings. Addressing this problem should be considered as future work (see Section 7.2.4), because doing so, S-Match will optimise its performance aggregating to the mappings discovered in the vanilla version, those discovered using the lexicon extension.

Although the number of true positives has reduced, the matcher discovers mappings that have not been output with the vanilla version. An example is:

- "*Speech sound disorder*" ≡ "*Phonological disorder*"

Thus, with the lexicon extension the matcher can interpret that "*speech sound*" and "*phonological*" are semantically related.

It is necessary to highlight that only considering equivalences, restricts the matcher to find "perfect mappings". This forces the extension to have the exact knowledge, because if something is considered more general (hypernym) or more specific (hyponym), the matcher will consider it a subsumption.

Regarding *f-measure*, the lexicon extension performs 6% worse than the baseline.

Grammar extension

The grammar extension increases more than 3% *recall*, mainly because of the postscript optimisations that simplifies labels. An example is the following one:

"*Schizophrenia*" $\equiv$ "*Schizophrenia, unspecified*".

After the postscript optimisation the labels are transformed into:

"*Schizophrenia*" $\equiv$ "*Schizophrenia~~, unspecified~~*".

Thus, a mapping that previously was considered as a subsumption is currently an equivalence.

Regarding *precision*, it slightly decreases 2%. In this case, it is a side effect of simplifying the labels, as the matcher outputs false positives that previously were discarded. An example is:

"*Intellectual disability (intellectual developmental disorder), Mild*" $\not\equiv$ "*Moderate intellectual disabilities*[3]"

After applying the grammar considerations the labels are transformed as follows:

"*Intellectual disability Mild*" $\not\equiv$ "*Moderate intellectual disabilities*"

Thus, the matcher outputs the mapping of both labels because in the KB "*mild*" and "*moderate*" are similar terms, and so for the matcher both labels are equivalent.

It is necessary to highlight that in this case the medical grammar rules included in the extension do not have any impact because they are mostly considered as subsumptions. For example:

"*Exhibitionistic disorder*" $\equiv$ "*Exhibitionism*"

[3]Disorders can be categorised as *mild*, *moderate* or *severe*. Therefore, in the medical domain, *mild* and *moderate* are not considered as synonyms.

In this case, even though "*exhibitionistic*" is included in the extension as a related form of "*exhibitionism*" the matcher will consider the source label as a specialisation of the target label, and so will output a subsumption semantic relation.

In terms of *f-measure*, this extension improves the baseline by 5%.

Lexicon and grammar extensions

In this case, the combination of both extensions makes that the matchers slightly improves the baseline in both *precision* and *recall*, around 1% and 3% respectively. Similarly as in the previous cases where extensions were used separately, S-Match does not have a significant improvement because most of the new mappings discovered with the extensions are output as subsumptions.

We can see how the lexicon extension contributes mainly reducing the number of false positives as it was explained above, whereas the grammar extension helps to find new true positives by simplifying labels.

The combination results in a kind of average aggregation of the individual performances of each extension. Thus, it is necessary to investigate in the future different ways of optimising these aggregations in order to obtain the best *precision* and *recall* when combining several extensions (see Section 7.2.4).

Regarding the *f-measure*, the combination of both extensions improves the baseline by around 4.5%.

6.5.1.2 Equivalences and Subsumptions

Table 6.2 and Figure 6.2 show the results of the experiment executed in S-Match considering as true positive both equivalences and subsumptions. We can see how the performance of S-Match improves its *precision* when using the medical extensions. However, S-Match vanilla version has the best performance in terms of *recall*.

The extensions enable the matcher to be more strict, being capable of discarding confusing mappings, and so, reducing significantly false positives, but there are also correct mappings which are rejected (e.g. the case of "*stereotypic movement disorder*" and the postscripts optimisation described above). Analysing the *f-measure*, we can conclude that DA matching improves the performance of S-Match by between 0.5%-20%.

Vanilla version

The most striking result from the vanilla version of S-Match (baseline, from this point on) is its low precision, which is caused by three causes. Firstly, considering the subsumption semantic relation entails that if the matcher finds this relation between any part of a source label and a target label, it outputs a mapping. This will discover more mappings than considering only the equivalence semantic relation, but this also produces more false positives, as it is described below with the use of the word "*state*". Secondly, the labels of the datasets to be matched correspond to medical descriptions which usually contains more than 4 or 5 words. S-Match transforms each label into a formula, so the more complex the label is the more complex the formula will be. Thirdly, these descriptions contain domain-specific knowledge which is not included into the matcher's KB, so it is not possible to match precisely this knowledge. Indeed, domain-specific grammar adds more complexity to the process because particular syntax and postscripts produce false positives. Below we can see some examples of false positives:

- "*Child affected by parental relationship distress*" $\not\sqsubseteq$ "*Mild cognitive impairment, so stated*"
- "*Problem related to living in a residential institution*" $\not\sqsubseteq$ "*Mild cognitive impairment, so stated*"
- "*Unspecified sleep-wake disorder*" $\not\sqsubseteq$ "*Obsessive-compulsive disorder, unspecified*"

In all of these cases, applicative grammar penalises the matcher for two main reasons:

1. Commas are used as disjunctive operators. Following this, the label formula corresponding to the first example is:

$$mild \,\&\, cognitive\ state \,\&\, impairment \mid state$$

   From this label formula S-Match computes the following node formula:

$$(mild \mid state) \,\&\, (cognitive\ state \mid state) \,\&\, (impairment \mid state)$$

That means that if "*state*" has a relationship with a lemma within any label of the other ontology, the matcher will output a mapping even if the rest of the label is not related.

2. Both "*state*" and "*unspecific*" are lemmas which are represented in the KB (WN) with a high degree of abstraction. This signifies that in a hierarchy organised by an *is-a* semantic relationship, they are located at the first levels, and so they have many hyponyms. For example, "*state*" is an inherited hypernym of "*distress*, "*disease*" and "*problem*", among others. Therefore it is likely that the matcher finds these semantic relations as these lemmas are common within the labels of the medical datasets that are used in the experiment (e.g. "*distress*" $\sqsubseteq$ "*state*").

In terms of *recall*, the baseline is the one which has the best performance. Mainly, the cause is completely related with the impact of applicative grammar as we have seen above. However, in this case, the matcher benefits from the way in which it handles grammar, getting the right mappings. Examples of true positives are:

- "*Mild neurocognitive disorder due to Alzheimer's disease*" $\sqsubseteq$ "*Mild cognitive impairment, so stated*"
- "*Mild neurocognitive disorder due to Parkinson's disease*" $\sqsubseteq$ "*Mild cognitive impairment, so stated*"

As a result, considering the general performance, *recall* is penalised by a lower *precision*, resulting in a *f-measure* around 42%.

Lexicon extension

In this case, we have enriched the matcher's KB with medical knowledge contained in SPECIALIST and MeSH. We can see how using domain (area of expertise) knowledge the number of false positives has reduced around 18%, but the number of true positives has also reduced about 12% so there is only a slight improvement of *precision* around 0.4% with respect to the baseline. The reason is that with this extension, the matcher has represented domain-specific knowledge that appears in some labels, therefore it can use this knowledge for the matching process, including them in the label formulas. Nonetheless, the number of these cases is small with respect to the whole dataset, resulting in a minimum improvement. Below there is an example of the true positives that have been found because of the medical knowledge provided by the lexicon extension:

- "*Other medication-induced parkinsonism*" $\equiv$ "*Other drug induced secondary parkinsonism*"

Thus, with the lexicon extension the matcher can interpret that "*medication-induced*" and "*drug induced*" are semantically related.

Regarding *recall*, we can see how with the extension, the number of true positives has reduced considerably, having a *recall* 5% worse than the baseline. The problem is caused because there are terms within the extension that do not have any other related form within the KB. This is the same reason as explained Section 6.5.1.1 with the example of "*Stereotypic movement disorder*" and "*Stereotyped movement disorders*".

Regarding *f-measure*, the lexicon extension only improves the baseline 0.4%.

Grammar extension

Regarding the grammar extension, the improvements in terms of *precision* are significant, enhancing this metric more than 17%. This result is mostly as a consequence of addressing applicative grammar (syntactic and postscript optimisations). Thus, this avoids undesired cases in which false positives were produced by the way in which S-Match works. For example, managing punctuation marks. An example, of false positive from the vanilla version and avoided with this extension is:

"*Unspecified sleep-wake disorder*" $\not\equiv$ "*Obsessive-compulsive disorder, unspecified*".

Thus, the number of false positives has drastically reduced more than 86% with respect to the vanilla version. In addition, there are also some mappings discovered by S-Match with this extension that were not output by the vanilla version. Below there are some examples of these true positives:

- "*Fetishistic disorder*" $\equiv$ "*Fetishism*"
- "*Hallucinogen persisting perception disorder*" $\equiv$ "*Hallucinogen use, unspecified with hallucinogen persisting perception disorder (flashbacks)*"

In the first case, it is domain grammar which plays a vital role as this mapping is discovered by the morphological expansion that we carried out following derivational morphology rules that are particular to the medical domain. The second case, is a clear example of how addressing applicative grammar (content within parentheses, commas...) helps S-Match to find correct mappings.

The *recall* is 12% worse than the baseline because the syntax and postscript optimisation, reduce the number of output mappings. That means that apart from drastically reducing the number of false positives caused, for example, by postscripts, there are also reduced the true positives that the matcher found as a side effect of using those postscripts.

In terms of *f-measure*, this extension improves the baseline by around 17%.

Lexicon and grammar extensions

The *precision* of this extension that combines both, lexicon and grammar extensions, gets the best of both worlds. It takes advantage of the domain-specific lexical knowledge and grammar. Thus, the extensions complement each other, reducing the number of false positives around 89.7% with respect to the baseline. As a result, this combined extension experienced the best performance in terms of *precision*, improving the baseline almost 22%.

In terms of *recall*, it is worse than the baseline at 12%, being also slightly worse than the *recall* obtained by the grammar extension. This is because of two main reasons. Firstly, S-Match with both, the lexicon and the grammar extensions finds mappings that are not discovered by the vanilla version, but there is some overlapping between both extensions. That means that some mappings are discovered in both extensions. For example:

- "*Social anxiety disorder (social phobia)* $\sqsubseteq$ "*Social phobia, unspecified*"

On the one hand, the lexicon extension includes "*social anxiety disorder*" and "*social phobia*" in the same synset, and so S-Match discovers this mapping when using the lexicon extension. On the other hand, the grammar extension addresses the parentheses and the postscript and, in this case, S-Match also outputs the mapping. Knowing that, we cannot expect that the matcher using the combination of both extensions will output the sum of those mappings discovered when it uses each extension individually, and that are not discovered in the vanilla version.

Secondly, in this case we have the same problem as when using the lexicon extension. There are mappings discovered by the matcher with the grammar extension that are rejected because a compound lemma has been included in the KB. An example is "*tobacco use disorder*":

- "*Tobacco use disorder, Mild* $\not\sqsubseteq$ "*Tobacco use*"

Regarding the *f-measure*, the combination of both extensions improves the baseline by almost 20%.

Conclusions

At this point, the results with S-Match suggest that our hypothesis is true, in general terms, as the matcher with DA functionality improves the baseline in *f-measure*. The only exception appears when the matcher uses the lexicon extension and only considers equivalences. However, we have seen how if S-Match does not consider both equivalences and subsumptions, it is penalised and cannot make the most of the extensions.

On the one hand, when the matcher only considers equivalences, it achieves a high *precision* in all cases, obtaining the best when the matcher uses the lexicon extension. Regarding *recall* it is pretty low in all cases, obtaining the best one the matcher with the grammar extension. The best performance in terms of *f-measure* is achieved when the experiments are executed with the grammar extension.

On the other hand, considering both equivalences and subsumption, *precision* is considerably improved with the grammar extension, obtaining the best performance when the two extensions are combined. This is evidence of the impact of grammar, domain and applicative, on matching domain ontologies. In our case it is applicative grammar which has the major impact. Regarding *recall*, the best performance is obtained by the vanilla version. The best performance in terms of *f-measure* is obtained by the combination of both lexicon and grammar extensions.

6.5.2 Logmap

LogMap differs from S-Match in the way in which it carries out the matching process (see Section 3.1.2.2) and in its KB, which does not includes WN. Thus, the two matchers output different results.

Table 6.3 and Figure 6.3 show the results of the experiment executed in LogMap. We can see that LogMap vanilla version has the best *precision*, however when the matcher uses the extensions, the *precision* does not get more than 3% worse. Regarding *recall*, any extension improves the baseline by around 2%. Analysing the *f-measure*, the extensions slightly improve the performance of LogMap by 2%.

Vanilla version

LogMap is conceived to output mappings with an acceptable confidence degree. Thus, firstly, it computes a set of possible mappings and secondly, it filters those which are considered as weak mappings. Knowing this, we can understand the high *precision* of the matcher being more than 55%. In addition, LogMap already uses SPECIALIST as KB, so it includes medical knowledge that is useful for matching the datasets of our experiment.

Regarding, *recall*, the value is less than 17,5%. In this case, there are two main reasons:

1. The matcher is configured with a high value for the minimum similarity degree threshold. The main reason of doing this is to avoid false positives, but this also produces that true positives, with a similarity degree below the threshold, cannot be output.

2. LogMap's KB only contains medical lexicon knowledge. In principle, we might think that this should be enough for matching datasets that contain medical descriptions. However, the main problem is that the length of descriptions is usually more than 5 words, which means that apart from medical terms, they normally contain other terms that represent general knowledge. Therefore, the lack of general knowledge has a negative impact on LogMap, not allowing it to discover more true positives.

In terms of *f-measure*, the *precision* compensates the low *recall*, resulting in a *f-measure* of 26.7%.

Lexicon extension

It is necessary to recall that this extension enriched WN with medical knowledge from SPECIALIST and MeSH. The main reason of doing this is because resources usually contain knowledge from the different levels of specificity. In our experiment, we can see this in the medical descriptions that are in the datasets.

Using this extension, LogMap not only has benefited from the medical knowledge, but also from the general knowledge that is included in WN. Actually, the medical knowledge that is new for the matcher is only the one from MeSH, as LogMap uses SPECIALIST by default KB.

The *precision* of this extension is around 1% worse than the baseline because the general knowledge contained in the extension produces that LogMap computes a higher similarity degree between some unrelated labels, producing more false positives. Examples of these false positives are:

- "*Problem related to living alone*" $\not\equiv$ "*Unrelated problems related to employment*"

- "*Narcolepsy without cataplexy but with hypocretin deficiency*" $\not\equiv$ "*Narcolepsy with cataplexy*"

In these cases, in the lexical indexation process [79], each entry is lexically expanded with general knowledge. This produces a higher similarity degree between the classes and so, the matcher outputs these mappings when using this extension. Despite these problems, *precision* is over 55%, which is still a good *precision* value compared with the baseline.

In terms of recall, the improvement with respect to the baseline is around 2%, discovering 10.85% more true positives. The reason for this is the use of general and domain knowledge which helps the matcher to find mapping of descriptions that include domain-specific terms, in our case, medical terms. An example of a true positive discovered by the medical knowledge included in the extension and that was not output by the vanilla version is:

- "*Idiopathic central sleep apnea*" $\equiv$ "*Primary central sleep apnea*"

Nonetheless, it is the general knowledge which has the major impact, discovering more mappings. Examples of these true positives are:

- "*Acute stress disorder*" $\equiv$ "*Acute stress reaction*"

- "*Amphetamine or other stimulant withdrawal*" $\equiv$ "*Other stimulant dependence with withdrawal*"

- “*Problem related to current military deployment status*” ≡ “*Military deployment status*”

The *f-measure* improves the baseline by 2%.

Grammar extension

Regarding *precision*, the grammar extension achieves 54.6%, which is 3% worse than the baseline. This is also caused by the false positives produced by the extension, which in this case contains WN with the morphological derivations.

The *recall* is 2% better than the baseline, with a value of 19%. LogMap benefits from WN with the morphological derivations, but in this case, medical grammar produces a minimal effect as LogMap carries out string similarity for each element in the label. Therefore, mappings such as “*Fetishism*” ≡ “*Fetishistic disorder*” are discovered by the vanilla version.

It is applicative grammar which has a slightly major impact, finding true positives not discovered by the vanilla version. Examples of these mappings are:

- “*Trichotillomania (hair-pulling disorder)*” ≡ “*Trichotillomania*”
- “*Panic disorder*” ≡ “*Panic disorder [episodic paroxysmal anxiety]*”
- “*Overweight or obesity*” ≡ “*Obesity, unspecified*”

In terms of *f-measure*, this extension obtains 28,5%, improving the baseline around 2%.

Lexicon and grammar extensions

Regarding *precision* we can see how it is 54.4%, being 3.2% worse than the baseline. In this case, the combined extension produces 602 false positives, which are the union of the false positives that LogMap outputs using the lexicon and the grammar extensions individually. In terms of *recall*, it improves the baseline 1,6%. Here we can see how combining the extensions the number of true positives is higher than the one that they perform individually. That means that some of the true positives identified by the extensions are different and now they are aggregated. Thus, the result is the union of the true positives discovered by the matcher when it performs the matching process using the lexicon and the grammar extensions separately.

The *f-measure* is more than 28%, improving the baseline in around 1.5%.

Conclusions

After analysing the results, they suggest that our hypothesis is true for LogMap. Thus, LogMap with DA functionality improves the baseline in terms of *f-measure*. However, DA functionality does not enhance *precision* and *recall*. The former has a slightly worse performance, but maintaining it over 55%, and the latter improves the baseline around 2%.

In this case, domain lexicon and applicative grammar have contributed to discover new mappings with respect to the vanilla version. However, it is their combination with general knowledge (WN) which have produced the major impact on the discovery of new mappings.

The best performance is carried out by the combination of both extensions (lexicon and grammar), because they complement each other by aggregating their individual results. However, this combination also produces the aggregation of the false positives output by the matcher when using the lexicon and the grammar extension separately. This has a negative impact on *precision*, so it is necessary to carry out research in techniques which optimise the combination of extensions maximising both, *precision* and *recall*.

6.6 Experiment 2.- Reduced DSM-5 Dataset

Overall, S-Match and LogMap have a low performance executing Experiment 1 (see Section 6.4), not reaching 30% of f-measure. The main cause of these results is provoked by the length of some of the labels within the datasets. The longer a label is the more likely is that the matcher does not discover the true positive. Moreover, these long labels also entails higher computational costs, requiring a powerful computer.

The aim of the experiment is to analyse the impact of our solution in a less complex dataset that also requires lower computational costs. To do so, we have selected a random dataset from DSM-5 with 200 descriptions. The only requirement for selecting a description was that it cannot contain more than 8 words. The main reason to do so, is to reduce the complexity of the dataset. Once we selected the source dataset, we developed a target dataset, selecting 177 descriptions from ICD-10 that are the correspondences of the source dataset. Similarly as in Experiment 1, we evaluate the results by comparing them with the gold standard, in which are defined 200 correspondences between both datasets.

6.6.1 Results

Below there are tabulated the *true positives, false positives* and *false negatives* obtained after running Experiment 2 with both S-Match and LogMap.

6.6.1.1 S-Match

Only Equivalences

	Total	True Positives	False Positives	False negatives
S-Match	200	47	1	153
S-Match (Lexicon extension)	200	48	1	152
S-Match (Grammar extension)	200	56	2	144
S-Match (Both extensions)	200	59	1	141

Table 6.4: Results experiment 2 with S-Match considering only equivalences

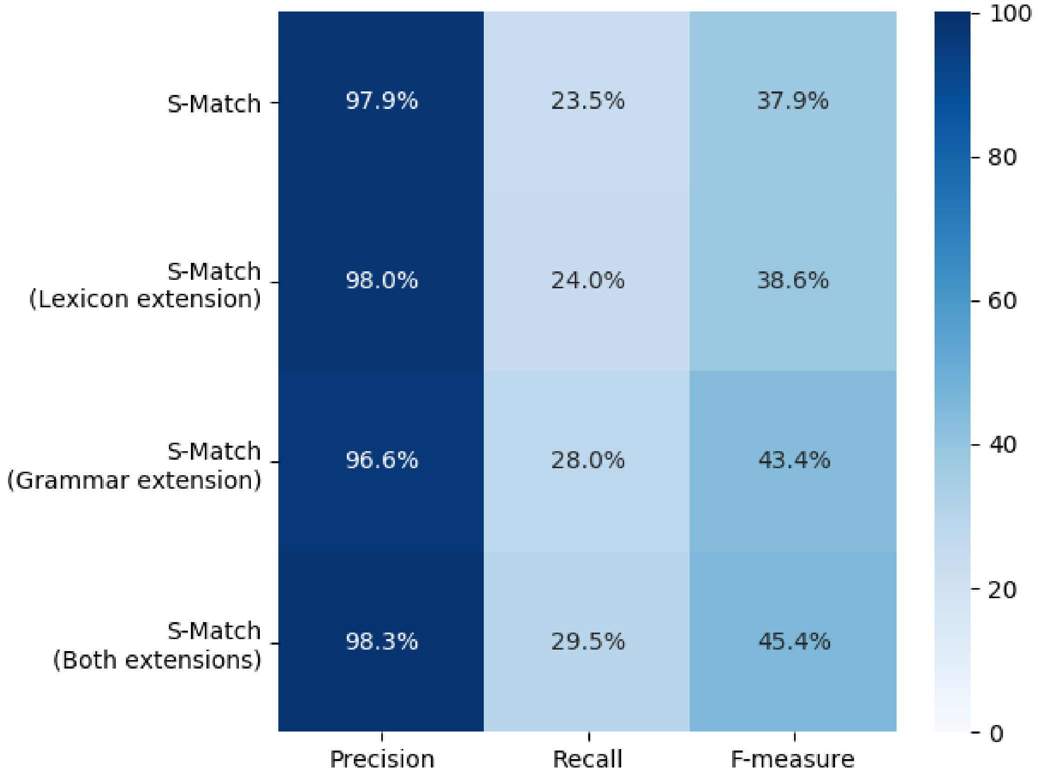

Figure 6.4: Metrics results experiment 2 with S-Match considering only equivalences

Equivalences and Subsumptions

	Total	True Positives	False Positives	False negatives
S-Match	200	113	241	87
S-Match (Lexicon extension)	200	102	98	98
S-Match (Grammar extension)	200	104	46	94
S-Match (Both extensions)	200	103	41	95

Table 6.5: Results experiment 2 with S-Match considering equivalences and subsumptions

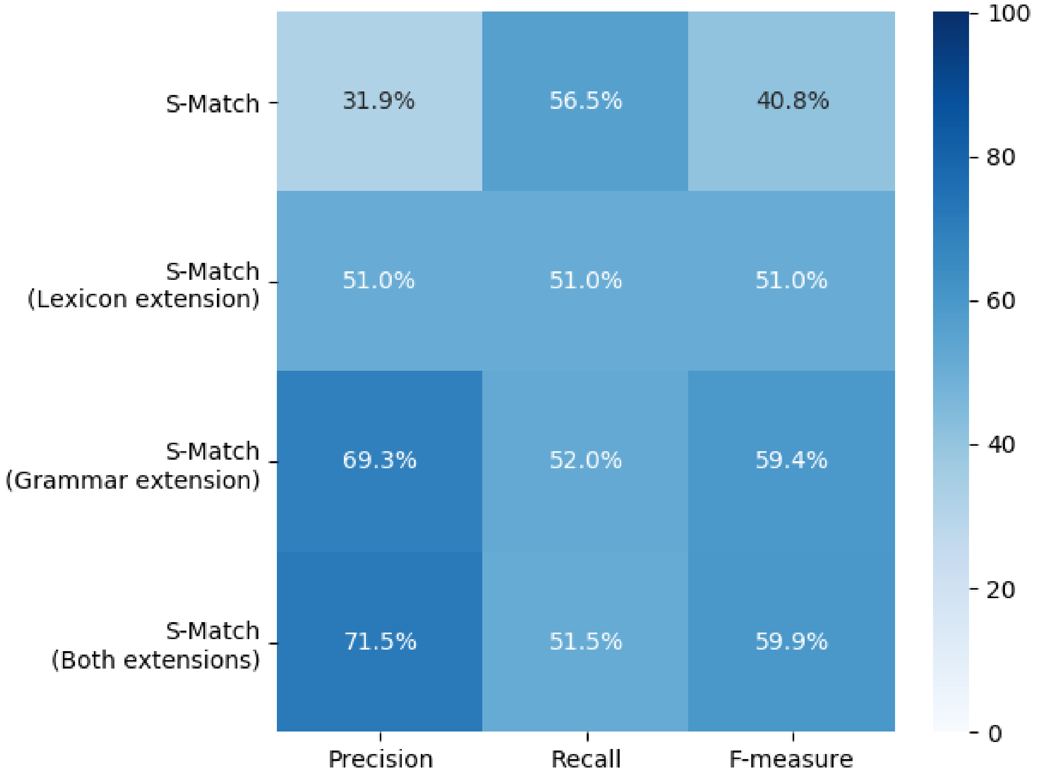

Figure 6.5: Metrics results experiment 2 with S-Match considering equivalences and subsumptions

6.6.1.2 LogMap

	Total	True Positives	False Positives	False negatives
LogMap	200	89	3	111
LogMap (Lexicon extension)	200	106	9	94
LogMap (Grammar extension)	200	106	8	94
LogMap (Both extensions)	200	108	9	92

Table 6.6: Results experiment 2 with LogMap

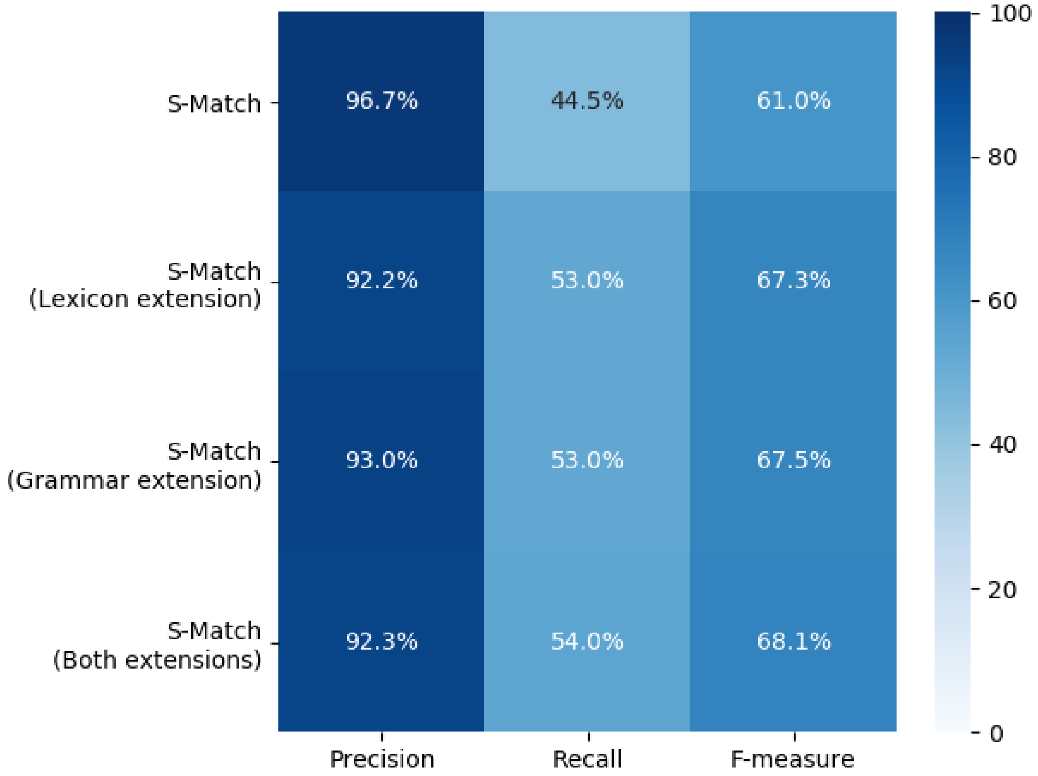

Figure 6.6: Metrics results experiment 2 with LogMap

6.7 Analysis Experiment 2

In general terms, we can see how both matchers improve their results with respect to Experiment 1 (see Section 6.4). Below we analyse the results obtained with each matcher.

6.7.1 S-Match

Tables 6.4 and 6.5, and Figures 6.4 and 6.5 show the results of S-Match.

6.7.1.1 Only Equivalences

We can see how *precision* is over 96% and *recall* is over 23% in all configurations. The main reason for these good results is that with less complex labels the matcher computes simpler label formulae which makes easier discovering true positives and reduces the number of false positives. As a result, all *f-measure* values are over 37%.

Regarding the extensions we can observe that all of them contribute to improve S-Match's performance. In particular, the configuration that uses both extensions, lexicon and grammar, has an improvement of 8% in *f-measure* with respect to the baseline.

6.7.1.2 Equivalences and Subsumptions

In this case, we can see a clear example of the impact of complex labels in S-Match's performance. Whereas in Experiment 1, the baseline has 5% *precision* and 42% *recall*, in this experiment *precision* is over 31% and *recall* over 56%. There is a drastic reduction of false positives, around 85%. Apart from the generalised improvement with respect to Experiment 1, we can observe the positive impact of each extension.

The lexicon extension improves *precision* around 20% because of two main reasons. Firstly, the extension provides new medical knowledge that helps the matcher to discover new true positives. Secondly, this new knowledge also contributes to reduce the number of false positives as we described in Section 6.4. This reduction of false positives also entails that some true positives associated to them are also removed. That means that *recall* is slightly reduced. Nonetheless, *f-measure* improves the baseline around 10%.

The grammar extension drastically reduces the number of false positives caused by the applicative grammar. This reduction, around 81% with respect to the baseline results in an improvement around 20% in terms of *f-measure*. In this case, the slight decrease of *recall* is compensated with the significant improvement in terms of *precision*.

The combination of both extensions is the configuration that obtains the best *precision*, 71.5%, and *f-measure*, around 60%. This is an improvement around 20% in terms of *f-measure* with respect to the baseline. We can see how both extensions complement each other to obtain the most reduced number of false positives in all configurations and discover new true positives. However, the grammar extension obtains a better *recall* than the combined extension. In the future, we will explore different aggregation techniques to optimise the combination of these results (see Section 7.2.4).

6.7.2 LogMap

Table 6.6 and Figure 6.6 show the results of Experiment 2 executed with LogMap.

We can see how LogMap's performance improves around 40% with respect to Experiment 1 (see Section 6.4). The main reason is that matchers in general, and LogMap in particular, are conceived to match ontology labels which usually do not have more than 3 or 4 words. Thus, matching medical descriptions that have more than 10 words, as we did in Experiment 1, challenges the matchers which output poor results.

In this case, all the extensions slightly improve the *f-measure* with respect to the baseline. The causes are the same as described for Experiment 1 (see Section 6.4), but here we can see a major impact.

The lexicon extension improves *recall* around 9%, mainly because of the general knowledge provided by WN and some medical knowledge from MeSH. The general knowledge also caused that the matcher found more *false positives*, resulting in a *precision* around 4% worse.

The grammar extension also helps to find new true positives, but it has lower impact than in S-Match, because LogMap carries out diacritic suppression, and label normalisation as a preprocessing step by default. Nonetheless, it slightly improves its performance 6.5% with respect to the baseline.

The combination of both extensions, obtains the best performance in terms of *f-measure*, improving the baseline more than 7%. This is the configuration in which LogMap discovers more true positives, resulting in the combination of the true positives obtained by the lexicon and grammar extensions separately. However, the false positives are also aggregated, negatively affecting *precision*, which is around 4% worse than the baseline. The optimisation of the aggregation process is considered as a future work (see Section 7.2.4).

6.7.3 Discussion

The analysis of the results produced by S-Match and LogMap suggests that DA matching enhance matchers' performance regarding their *f-measure*. Therefore, we can conclude that it is a recommendable option to integrate domain-knowledge extensions into matchers' KB. Despite the benefits that we have seen in terms of performance, we have also identified several limitations that are necessary to be considered:

- *Extensions might be incomplete.* We have to assume that the whole knowledge of a particular domain cannot be represented in an extension.

That means that there might be knowledge in the labels to be matched that is not included in the extensions. Knowing that, it is necessary to find a balance between the cost and benefit of an extension, deciding which knowledge is the most representative. Otherwise, trying to be exhaustive will extremely increase the cost, reporting in return an insignificant benefit.

- *Generating extensions is not trivial.* As detailed in chapters 4 and 5 the generation of domain-specific extensions is a complex process that usually involves several experts who have to reach an agreement about how to represent knowledge. Specially if the extension involves the formalisation of unstructured data. However, it is recommendable the generation of extensions when it is necessary to manage resources that extensively contain domain-knowledge. Depending on the needs and the resources to be matched some extensions are more recommendable than others. For example, if our priority is having a high *recall*, the extension should contain domain and applicative grammar, and general knowledge. In case of prioritising *precision*, a lexicon extension with general, area of expertise and applicative knowledge is the best choice.

- *Difficult to maintain.* The evolution of language and the discovery of scientific breakthroughs make constant the representation of new knowledge. Updating an extension with this new knowledge usually implies more than just appending it, requiring in most cases carrying out the whole process of creating an extension.

- *Grammatical Patterns.* In our research we have analysed medical grammar manually, by delving into medical resources. However, this is not recommendable because it implies a huge effort and consumes a lot of time. A possible solution might be the application of pattern recognition algorithms to discover the use of grammar that is typical in a particular domain.

- *Diversity of applicative grammar.* At the applicative level of specificity it is likely that different resources and organisations define their particular grammatical conventions. This produce a wide range of applicative grammars. In addition, at this level there are less resources than in a more abstract level of specificity (e.g. The Mayo clinic has less resources than the medical domain in general). This means that if there is not a large number of documents, it is not possible to apply machine learning approaches such as pattern recognition algorithms.

In our experiment, we have included an execution in which the matcher was enriched with two extensions, the lexicon and the grammar ones. The performance in terms of *f-measure* was the best in both matchers, so that, the extensions are complementary and their combination is recommendable. Nonetheless, we have also found several limitations:

- *Fusion techniques.* Currently, matchers aggregate their results, but there are cases in which the combination produces a result slightly worse than when the extension is used individually. For example, the lexicon extension in S-Match produces a better *recall* than when the matcher uses grammar and lexicon extensions together. Similarly, in LogMap, the lexicon extension has a better *precision* than the combined extension. For this reason, it is necessary to investigate methods that allow the maximisation of both *precision* and *recall*, when extensions are combined (see Section 7.2.4).

Another aspect that we have realised during the analysis of the results is that S-Match outputs some mappings which may "*improve*" the gold standard. Identifying these cases is interesting for clinicians who have enough expertise to change the gold standard, incorporating these alignments in new versions. The "*correct*" correspondences that are found by the matcher, but are not included in the gold standard are:

- Other specified insomnia disorder $\sqsubseteq$ Sleep disorder, unspecified
- Other specified insomnia disorder $\sqsubseteq$ Other sleep disorders
- Specific phobia, Fear of other medical care $\sqsubseteq$ Other specified phobia
- Idiopathic central sleep apnea $\sqsubseteq$ Sleep disorder, unspecified
- Tobacco use disorder, Moderate $\sqsubseteq$ Tobacco use
- Tobacco use disorder, Severe $\sqsubseteq$ Tobacco use
- Dissociative amnesia with dissociative fugue $\sqsubseteq$ Dissociative amnesia

- Central sleep apnea comorbid with opioid use $\sqsubseteq$ Sleep disorder, unspecified
- Adjustment disorder, Unspecified $\sqsubseteq$ Adjustment disorder with anxiety
- Language disorder $\sqsubseteq$ Other developmental disorders of speech and language
- Narcolepsy with cataplexy but without hypocretin deficiency $\sqsubseteq$ Sleep disorder, unspecified
- Obstructive sleep apnea hypopnea $\sqsubseteq$ Sleep disorder, unspecified
- Bipolar II disorder $\sqsubseteq$ Bipolar disorder, unspecified
- Schizotypal personality disorder $\equiv$ Schizoid personality disorder
- Circadian rhythm sleep-wake disorders $\sqsubseteq$ Circadian rhythm sleep disorder, delayed sleep phase type
- Circadian rhythm sleep-wake disorders $\sqsubseteq$ Circadian rhythm sleep disorder, advanced sleep phase type
- Circadian rhythm sleep-wake disorders $\sqsubseteq$ Circadian rhythm sleep disorder, irregular sleep wake type
- Anorexia nervosa, Binge-eating/purging type $\sqsubseteq$ Binge eating disorder
- Obsessive-compulsive and related disorder due to another medical condition $\sqsubseteq$ Obsessive-compulsive disorder
- Adjustment disorder, With anxiety $\sqsubseteq$ Adjustment disorder, unspecified
- Narcolepsy without cataplexy but with hypocretin deficiency $\sqsubseteq$ Sleep disorder, unspecified

Discussing the correctness of these correspondences with different health professionals of the Andalusian Health Service (SAS), they agreed that these correspondences are correct and they can be added to the gold standard. However, in order to be coherent with the correspondences defined in DSM-5, we have not enriched the gold standard with these mappings, even knowing that S-Match is currently unfairly penalised.

6.8 Summary

In this chapter, we have evaluated the hypothesis of our research, suggesting that domain information helps to improve the performance of two of the most popular matchers, S-Match and LogMap, which perform the matching process following really different approaches. Moreover, DA-matchers discover new mappings that otherwise cannot be found. Both matchers improve their performance in terms of *f-measure* with respect to the baseline. LogMap around 2% and 7%, and S-Match around 20% in experiments 1 and 2, respectively. However, *precision* and *recall* do not improve together. In S-Match *precision* considerably improves and *recall* slightly decreases, whereas in LogMap, *recall* slightly improves and *precision* is marginally worse. S-Match gets the best performance when it combines lexicon and grammar extensions, whereas LogMap obtains it, using the lexicon extension in Experiment 1 and the combined extension in Experiment 2. Finally, we discussed several limitations that we have detected and that are recommendable to be addressed in the future.

Chapter 7

Conclusion

In this chapter, we summarise the research presented in the thesis. Firstly, we present the concluding remarks that have been extracted from our research, highlighting the encountered benefits and limitations. Then, there are considered several possible future works, in order to address the current limitations or to apply our solution to other scenarios. The chapter finishes with the resources and research papers that we have produced during the PhD.

7.1 Concluding Remarks

In this thesis, we have focussed on addressing problems that appear in current OM approaches, which negatively affect matchers' performance in terms of *precision* and *recall*. Thus, we formulated the following hypothesis:

"*When matching domain ontologies, matchers with DA functionality*[1] *have a better performance in terms of precision and recall than those which do not have this functionality because domain knowledge helps them to disambiguate and discover mappings that otherwise could not be found, and reject mismatches that look superficially plausible.*"

In order to prove this hypothesis, we have carried out research in this direction, producing as a result the following contributions:

- *Novel conception of domain-knowledge.* We have presented an innovative way of analysing domain-knowledge by considering three different dimensions: *specificity* (degree of knowledge specialisation), *language*

[1]For DA functionality we mean the knowledge and considerations taken by matchers to carry out the OM process in each particular domain.

structure (role of lexicon and grammar) and *type of knowledge resources* (regarding generation methodologies).

- *New classification of domain-knowledge resources.* We have presented a classification in which all domain-knowledge resources can be classified attending to the combination of the three dimensions of domain-knowledge.

- *Several approaches using domain-knowledge.* In order to allow matchers taking advantage of domain-knowledge we have presented different approaches which can be used depending on the data that we have and the requirements of the scenario in which the OM process will be applied. Prior to outline the approaches, it is necessary to highlight that after our analysis in ER and medical resources we have realised that within a knowledge resource, normally there is represented knowledge from the different levels of *specificity*. For example, within a medical description of a disease, apart from terms from the medical domain there are also terms that represent general knowledge. For this reason, we decided to integrate domain-specific knowledge within a domain-independent resource (WN) with the aim that matchers can take advantage of knowledge from different levels of *specificity*.

  – *Integrating symbolic resources.*

    1. *Fine integration.* This is an approach that allows the integration into WN of all kinds of data (structured and unstructured). Mainly, for each new term that is going to be integrated, it is necessary to find in which place of WN's hierarchy it fits better, considering the semantic relations that it has with the existing synsets. In our case, we have applied this methodology to produce an ER extension for WN, from terms included in the UKCP lexicon. However, this is a rudimentary methodology which involves a huge effort and supervision, so it should only be used when there is no other option due to the nature of data or ontology engineering constraints which require a precise and adapted integration.
    2. *Rough integration.* This approach takes advantage of structured knowledge and extract a sub-tree to plug it into WN. In this

case, we have to identify the place at which the root node of the sub-tree should be located and the semantic relations that it has with the synsets within WN. We followed this methodology to integrate MeSH and Specialist into WN. This method is carried out semi-automatically, so the integration requires less effort than the *fine integration*. Nonetheless, this integration is less precise, as the only element which is fully integrated with all semantic relations within WN elements is the sub-tree root node, so all the other sub-tree elements will only have the inherited semantic relations from the root node and the semantic relations that they had defined in the source resource.

Regarding the application domains that we have used in our research, the requirements between ER and the medical domain are completely different. In the former, there might be cases in which the agencies that participate in an ER scenario have not collaborated before, so the alignment of their resources should be carried out as soon as possible. Therefore, in these cases it is recommendable the use of the rough integration methodology as it is faster than the fine integration one. We can venture that this will be possible in a close future as agencies are working in producing structured data resources, but with the current resources that include mostly unstructured data, the only option is that the alignment between resources has to be carried out in a preparation phase prior to the triggering of an ER scenario.

As regards the medical domain, time is not a problem, but in this case it is also recommendable following the rough integration as we have done. Mainly, because there are available many structured and high quality resources which contain the semantics that experts in the field agreed. This means that even though only integrating a sub-tree of this resource into WN, the domain-specific semantics will be sound.

– *Integrating grammar*. This novel approach aims at enriching matchers' KB with a morphological expansion, following derivation morphology rules defined in each domain. In our case, we have produced an extension that follows 14 derivation morphology rules of the medical domain. This extension, which contains around 9,000 derivations, was integrated into WN.

Apart from that, this grammar approach also involves a syntactic and postscript optimisation, in order to avoid matchers to be penalised by the unstandardised use of syntax and postscripts within domain-knowledge resources.

The approaches have been evaluated by two experiments which consisted of matching two classifications of diseases, with two different matchers: S-Match and LogMap. Each experiment has been executed four times for each matcher with the following configurations: matcher's vanilla version, vanilla version plus lexicon extension, vanilla version plus grammar extension, and vanilla version plus both extensions (lexicon and grammar). The results suggest that our hypothesis is true considering that there is an improvement in both matchers with respect to their *f-measure*. Therefore, domain-knowledge improves the performance of matchers that take advantage of it. Nonetheless, even improving *f-measure*, *precision* and *recall* do not improve at the same time. Indeed, when *recall* increases, *precision* slightly decreases, and the other way round.

Despite having positive results, we have found several limitations that need to be addressed in the future. Examples of these limitations are: extension incompleteness, difficulty to generate and maintain extensions, finding grammatical patterns associated to a domain, difficulties to deal with domain-knowledge at applicative level and fusion techniques between extensions (see Section 6.7.3).

In conclusion, the results suggest that our hypothesis is true in terms of *f-measure*, but there are certain issues that still need to be addressed.

7.2 Future Work

This new conception of domain-knowledge opens a wide scope of new possibilities in which matchers can benefit from domain-specific BK. Our contribution and the defined approaches should be considered as a starting point for new proposals and methodologies. Below we define several possible research lines that may be followed to move forward or address the limitations of the research described in this thesis.

7.2.1 Instantiating our Approaches in other Matchers

Once we have seen that our approaches enhance matchers' performances, it is interesting to integrate our extensions into the most relevant matchers that participate in the OAEI. For example, AML is the matcher which has the best performance, so it is interesting to see how much impact will have our approaches on improving its performance.

Having integrated the top OAEI matchers with DA functionality, this can allow organisations to choose the matcher which performs better in terms of *precision, recall*, or *time*[2], depending on the needs of each scenario.

7.2.2 Domain Adaptation

So far, we are considering that matchers have domain-awareness as they take advantage of knowledge that is specific of a particular domain. In our case, because we are working in the medical domain, the matcher just focuses on lexicon and grammar from the medical domain. However, in order to make this functionality extensible to other domains, it might be interesting to add to the matcher the functionality to detect the domain of the ontologies to be matched and adapt to it by selecting the appropriate resources that belong to each particular domain.

The Diversicon framework can be really useful in this case, because domain-knowledge extensions can be stored in its repository. Thus, if a matcher detects that it is going to match two ontologies from the Architecture domain, it can adapt to this domain by automatically choosing the resources from the Architecture domain that are available on the Diversicon repository. That means that matchers might select the most appropriate KB (enriched with specific domain-knowledge) for carrying out the matching problem, depending on the domain.

7.2.3 Next Steps on Domain-Knowledge Grammar

The next steps to follow with respect to domain-knowledge grammar should focus on its automatic detection in different domains. On the one hand, this

[2]In our research we have not considered time for evaluation purposes. Nonetheless, in the future it might be interesting having a ranking of matchers regarding different metrics, and time might be one of them.

involves the identification of grammar rules that allow morphological derivation as the ones that we integrated in our grammar extension. On the other hand, domain-knowledge resources might be automatically analysed by applying pattern recognition algorithms, with the aim of finding possible postscripts, domain collocations and particular uses of syntax. The main requirement for carrying out this process is that a large number of domain resources need to be available for applying these machine learning techniques. As suggested before, the Diversicon repository can store these extensions. In this case, grammar extensions and the different optimisation tasks.

Apart from that, it might be useful the development of a tool which provides matchers with all tasks such as syntactic and postscript optimisation which cannot be integrated into matchers as extensions[3]. Thus, the only requirement will be the integration of this tool into each matcher. Therefore, when a matching process takes place, the tool can carry out the optimisation tasks that requires the domain of the ontologies to be matched, selecting them automatically from Diversicon's repository.

7.2.4 Fusion techniques

We have seen how in S-Match, the combination of different domain-knowledge extensions produces a better performance in terms of *f-measure* than when matchers use these extensions individually. However, this does not occur with LogMap in Experiment 1, so it is necessary to investigate which extensions should combine and in which cases it is recommendable combining them or not.

Moreover, currently, when two extensions are combined they aggregate their results. This produces that some times the *precision* or *recall*, of the matcher's performance when using only one extension is better than when it combines different extensions. For this reason, it is necessary to study which is the best workflow for obtaining the best matcher performance that maximises both, *precision* and *recall*. Thus, it might be advisable the execution of the matcher with one extension and after that, using the output to carry out a second execution with another extension. In this case, the mappings obtained in each extension might have a different weight, according to the importance of each extension, that can be used in the process of aggregating results.

[3]Currently, these tasks are executed in a preprocessing phase prior to the matching process.

7.2.5 Application to Emergency Response

As it was explained in Section 3.3.3, we had to abandon our research in the ER domain due to the lack of available and accessible resources to evaluate our approaches. However, speculating that these resources will be available, the solution presented in this thesis can be integrated into CHAIn [105] to allow query rewriting between agencies from different areas of expertise.

Currently, we have an ER extension with terms from the UKCP glossary. Nonetheless, it is also interesting the development of a grammar extension. To do so, it is necessary to explore whether there are grammar rules specific to the ER and to find patterns of applicative grammar within the resources to be matched[4].

Following the ideas proposed by Bella et al. [14], our solution might be also applicable to cross-border ER scenarios, that involve agencies which speak different languages.

7.2.6 Ontology Matching in Consumer Health

The concept of "*Consumer Health (CH)*" [39, 88] references the use case in which people take advantage of the web to investigate and try to understand their diseases or symptoms [40, 41]. However, even though they have access to many resources, usually they do not understand the information that they find [31] because medical terms are too technical, and so, difficult to be known and understood by non-experts [87]. This problem has existed in daily patient-doctor communication because laypersons and medical experts differ in their KR, it being in some cases impossible to match lay terms with medical ones [83]

Recently, researchers have developed several resources with lay medical terms trying to bridge the gap between non-experts and experts. Zeng and Tse developed a first generation of CH vocabularies with the aim of bridging lay forms, concepts and relations with the medical domain [150]. They realised that consumer-formed query terms often do not match with technical medical resources, such as the UMLS [20]. Examples are *Coronavirus* with *COVID-19*, *belly button* with *navel* or *weight loss surgery* with *bariatric surgery*. Applying our solution and enriching systems with this domain-knowledge coming from both, health professionals and patients will help to address the mentioned

[4]The UKCP glossary can be used to find applicative grammar as this resource integrates entries from different organisations, which follows specific syntactic standards.

misunderstanding problems.

7.2.7 Multilingual Biomedical Ontology Matching

Our solution can be also applied to address one of the challenges proposed in the 2nd Workshop on Multilingual Biomedical Text Processing, which focuses on how dealing with localization issues, including adaptation to local varieties of international languages (UK vs US English, Spanish from Spain and Latin America or US).

This is another example in which we can apply our solution, which incorporates medical domain-knowledge, and the ideas presented by Bella et al. [14] to carry out language and DA OM.

It is necessary to highlight that multilingual OM, entails adapting grammar extensions to each language as the use of suffixes, postscripts and punctuation marks will differ between languages.

7.3 Contributions

Below the are outlined the resources[5] and the publications produced during the PhD.

7.3.1 Resources

- **F. J. Quesada Real.** Emergency Response extension for WordNet (100 terms from the UKCP lexicon), 3, 2017.
- **F. J. Quesada Real.** MeSH extension for WordNet (Disorders [C] and Psychiatry and Psychology [F]), 2, 2018.
- **F. J. Quesada Real.** SPECIALIST extension for WordNet, 3, 2018.
- **F. J. Quesada Real.** Medical Grammar extension for WordNet, 7, 2019.

7.3.2 Publications

- **F. J. Quesada Real**, F. McNeill, G. Bella, and A. Bundy. Improving dynamic information exchange in emergency response scenarios. *In*

[5]Resources are available on the link: https://github.com/s1580097/resources

Proceedings of 14th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2017), pages 824-833, 2017.

- **F. J. Quesada Real**, F. McNeill, G. Bella, and A. Bundy. Identifying semantic domains in emergency scenarios. In *15th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2018)*, pages 1130-1132, 2018.

- G. Bella, F. McNeill, D. Leoni, **F. J. Quesada Real**, and F. Giunchiglia. Diversicon: Pluggable lexical domain knowledge. *Journal on Data Semantics*, pages 1-16, 2019.

- **F. J. Quesada Real**, G. Bella, F. McNeill and A. Bundy. Using Domain Lexicon and Grammar for Ontology Matching. *International Semantic Web Conference 2020 - ISWC'20 - Workshop on Ontology Matching (OM-2020). Accepted.*

7.4 Summary

In this chapter, we reviewed the research included in the thesis. First of all, there are presented the concluding remarks showing the novelty of the proposed solution. Its benefits and the limitations have been highlighted next. After that, we pointed out several future directions than can be followed to continue researching in domain-knowledge matching. The chapter has concluded with the resources and publications that were produced in our research. In this chapter, we summarise the research presented in the thesis. Firstly, there are introduced some related works. After that, we present the concluding remarks that have been extracted from our research, highlighting the encountered benefits and limitations. Then, there are considered several possible future works, in order to address the current limitation or to apply our solution to other scenarios. The chapter finishes with the resources and research papers that we have produced during the PhD.

Bibliography

[1] G. Acampora, H. Ishibuchi, and A. Vitiello. A comparison of multi-objective evolutionary algorithms for the ontology meta-matching problem. In *2014 IEEE Congress on Evolutionary Computation (CEC)*, pages 413–420. IEEE, 2014.

[2] G. Acampora, V. Loia, S. Salerno, and A. Vitiello. A hybrid evolutionary approach for solving the ontology alignment problem. *International Journal of Intelligent Systems*, 27(3):189–216, 2012.

[3] Z. Aleksovski, M. Klein, W. Ten Kate, and F. Van Harmelen. Matching unstructured vocabularies using a background ontology. In *Managing Knowledge in a World of Networks*, pages 182–197. Springer, 2006.

[4] A. Algergawy, M. Cheatham, D. Faria, A. Ferrara, I. Fundulaki, I. Harrow, S. Hertling, E. Jiménez-Ruiz, N. Karam, A. Khiat, et al. Results of the ontology alignment evaluation initiative 2018. In *Proceedings of the 13th International Workshop on Ontology Matching*, pages 76–116, 2018.

[5] A. Algergawy, M. Cheatham, D. Faria, A. Ferrara, I. Fundulaki, I. Harrow, S. Hertling, E. Jiménez-Ruiz, N. Karam, A. Khiat, et al. Results of the ontology alignment evaluation initiative 2019. In *Proceedings of the 14th International Workshop on Ontology Matching*, pages 46–85, 2019.

[6] D. Allemang and J. Hendler. *Semantic web for the working ontologist: effective modeling in RDFS and OWL*. Elsevier, 2011.

[7] J. Allen. *Natural language understanding*. Pearson, 1995.

[8] S. Amarel. An approach to heuristic problem-solving and theorem proving in the propositional calculus. *Systems and Computer Science*, pages 131–171, 1967.

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

[10] A. P. Association et al. Diagnostic and statistical manual of mental disorders. *BMC Med*, 17:133–137, 2013.

[11] R. Baeza-Yates and B. Ribeiro-Neto. *Modern information retrieval*, volume 463. ACM press New York, 1999.

[12] D. J. Barnett, C. B. Thompson, N. L. Semon, N. A. Errett, K. L. Harrison, M. K. Anderson, J. L. Ferrell, J. M. Freiheit, R. Hudson, and M. McKee. Eppm and willingness to respond: the role of risk and efficacy communication in strengthening public health emergency response systems. *Health communication*, 29(6):598–609, 2014.

[13] S. Bechhofer, F. Van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider, and L. A. Stein. Owl web ontology language reference. *W3C recommendation*, 10(02), 2004.

[14] G. Bella, F. Giunchiglia, and F. McNeill. Language and domain aware lightweight ontology matching. *Journal of Web Semantics*, 43:1–17, 2017.

[15] G. Bella, F. McNeill, D. Leoni, F. J. Quesada Real, and F. Giunchiglia. Diversicon: Pluggable lexical domain knowledge. *Journal on Data Semantics*, pages 1–16, 2019.

[16] G. Bella, A. Zamboni, and F. Giunchiglia. Domain-based sense disambiguation in multilingual structured data. In *The Diversity Workshop at the European Conference on Artificial Intelligence (ECAI)*, pages 53–61, 2016.

[17] L. Bentivogli, A. Bocco, and E. Pianta. Archiwordnet: integrating wordnet with domain-specific knowledge. In *Proceedings of the 2nd International Global Wordnet Conference*, pages 39–47, 2004.

[18] L. Bentivogli, P. Forner, B. Magnini, and E. Pianta. Revising the wordnet domains hierarchy: semantics, coverage and balancing. In *Proceedings of the Workshop on Multilingual Linguistic Ressources*, pages 101–108. Association for Computational Linguistics, 2004.

[19] P. Bille. A survey on tree edit distance and related problems. *Theoretical computer science*, 337(1):217–239, 2005.

[20] O. Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(1):267–270, 2004.

[21] O. Bodenreider, T. F. Hayamizu, M. Ringwald, S. De Coronado, and S. Zhang. Of mice and men: Aligning mouse and human anatomies. In *AMIA Annual Symposium Proceedings*, volume 2005, page 61. American Medical Informatics Association, 2005.

[22] P. Bouquet, L. Serafini, and S. Zanobini. Semantic coordination: a new approach and an application. In *International semantic web conference*, pages 130–145. Springer, 2003.

[23] S. E. Bowman. Coordination of snomed-ct and icd-10: getting the most out of electronic health record systems. *Perspectives in Health Information Management*, 2005.

[24] R. J. Brachman and H. J. Levesque. *Readings in knowledge representation.* Morgan Kaufmann Publishers Inc., 1985.

[25] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, and F. Yergeau. Extensible markup language (xml). *World Wide Web Journal*, 2(4):27–66, 1997.

[26] R. Briggs. Knowledge representation in sanskrit and artificial intelligence. *AI magazine*, 6(1):32–32, 1985.

[27] A. Bundy, L. Byrd, G. Luger, C. Mellish, and M. Palmer. Solving mechanics problems using meta-level inference. In *Proceedings of the 6th International Joint Conference on Artificial Intelligence - Volume 2*, page 1017–1027, 1979.

[28] B. Chiu, G. Crichton, A. Korhonen, and S. Pyysalo. How to train good word embeddings for biomedical nlp. In *Proceedings of the 15th workshop on biomedical natural language processing*, pages 166–174, 2016.

[29] P. Chocron, D. Gromann, and F. J. Q. Real. Towards building ontologies with the wisdom of the crowd. In *Proceedings of Diversity ECAI 2016*

International Workshop on Diversity-Aware Artificial Intelligence, pages 1–11, 2016.

[30] P. Chocron and M. Schorlemmer. Ontology alignment evaluation in the context of multi-agent interactions. In *Proceedings of the 11th International Workshop on Ontology Matching, co-located with the 15th International Semantic Web Conference (ISWC)*, pages 25–36. CEUR Workshop Proceedings, 2016.

[31] V. Claveau, T. Hamon, S. Le Maguer, and N. Grabar. Health consumer-oriented information retrieval. In *Medical Informatics Europe conference, MIE*, page 80. IOS Press, 2015.

[32] M. Clinic. Glossary - science of health care delivery: Robert d. and patricia e. kern center. https://www.mayo.edu/research/centers-programs/robert-d-patricia-e-kern-center-science-health-care-delivery/about/glossary, 2020. [Online; accessed 14-January-2020].

[33] N. S. W. L. C. Commissioning. Glossary of terms. https://www.swlondon.nhs.uk/wp-content/uploads/2014/10/SWLCC-NHS-ACROYNM-GLOSSARY-2014-10-16-ClTh-v.02.pdf, 2014. [Online; accessed 21-January-2020].

[34] W. W. W. Consortium. Extensible markup language (xml) 1.1. 2006.

[35] O. Corcho, M. Fernández-López, and A. Gómez-Pérez. Methodologies, tools and languages for building ontologies. where is their meeting point? *Data & knowledge engineering*, 46(1):41–64, 2003.

[36] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.

[37] K. Donnelly. Snomed-ct: The advanced terminology and coding system for ehealth. *Studies in health technology and informatics*, 121:279, 2006.

[38] J. Euzenat and P. Shvaiko. *Ontology matching*, volume 18. Springer, 2007.

[39] G. Eysenbach. Consumer health informatics. *BMJ*, 320(7251):1713–1716, 2000.

[40] G. Eysenbach and A. R. Jadad. Evidence-based patient choice and consumer health informatics in the internet age. *Journal of medical Internet research*, 3(2):e19, 2001.

[41] G. Eysenbach and C. Köhler. How do consumers search for and appraise health information on the world wide web? qualitative study using focus groups, usability tests, and in-depth interviews. *Bmj*, 324(7337):573–577, 2002.

[42] F. Farazi, C. Chapman, P. Raju, and L. Melville. Wordnet powered faceted semantic search with automatic sense disambiguation for bioenergy domain. In *2016 IEEE Tenth International Conference on Semantic Computing (ICSC)*, pages 112–115. IEEE, 2016.

[43] D. Faria, C. Pesquita, E. Santos, M. Palmonari, I. F. Cruz, and F. M. Couto. The agreementmakerlight ontology matching system. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pages 527–541. Springer, 2013.

[44] C. Fellbaum. *WordNet*. Wiley Online Library, 1998.

[45] M. Fernández-López, A. Gómez-Pérez, and N. Juristo. Methontology: from ontological art towards ontological engineering. In *AAAI-97 Spring Symposium Series*, pages 33–40. American Asociation for Artificial Intelligence, 1997.

[46] A. N. C. for Classification in Health. International classification of diseases, 10th revision, australian modification. https://www.ihpa.gov.au/what-we-do/icd-10-am-achi-acs-current-edition, 2019. [Online; accessed 22-January-2020].

[47] G. Francopoulo. Lmf iso 24613:2008, Mar. 2008.

[48] G. Francopoulo. *LMF Lexical Markup Framework*. Wiley Online Library, 2013.

[49] A. Gal and P. Shvaiko. Advances in ontology matching. In *Advances in Web Semantics I*, pages 176–198. Springer, 2009.

[50] W. A. Gale, K. W. Church, and D. Yarowsky. One sense per discourse. In *Proceedings of the workshop on Speech and Natural Language*, pages 233–237. Association for Computational Linguistics, 1992.

[51] S. Gella, C. Strapparava, and V. Nastase. Mapping wordnet domains, wordnet topics and wikipedia categories to generate multilingual domain specific resources. In *LREC*, pages 1117–1121, 2014.

[52] F. Giunchiglia. Contextual reasoning. *Epistemologia, special issue on I Linguaggi e le Macchine*, 16:345–364, 1993.

[53] F. Giunchiglia and B. Dutta. Dera: A faceted knowledge organization framework. Technical report, University of Trento, 2011.

[54] F. Giunchiglia, B. Dutta, V. Maltese, and F. Farazi. A facet-based methodology for the construction of a large-scale geospatial ontology. *Journal on data semantics*, 1(1):57–73, 2012.

[55] F. Giunchiglia, V. Maltese, and B. Dutta. Domains and context: first steps towards managing diversity in knowledge. *Web Semantics: Science, Services and Agents on the World Wide Web*, 12:53–63, 2012.

[56] F. Giunchiglia, F. McNeill, M. Yatskevich, J. Pane, P. Besana, and P. Shvaiko. Approximate structure-preserving semantic matching. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pages 1217–1234. Springer, 2008.

[57] F. Giunchiglia and P. Shvaiko. Semantic matching. In *Knowledge Engineering Review Journal*, volume 18, pages 265–280, 2003.

[58] F. Giunchiglia, P. Shvaiko, and M. Yatskevich. S-match: an algorithm and an implementation of semantic matching. In *European semantic web symposium*, pages 61–75. Springer, 2004.

[59] B. Golden. Shortest-path algorithms: A comparison. *Operations Research*, 24(6):1164–1168, 1976.

[60] A. González, G. Rigau, and M. Castillo. A graph-based method to improve wordnet domains. In *Computational Linguistics and Intelligent Text Processing*, pages 17–28. Springer, 2012.

[61] A. Gonzalez-Agirre, M. Castillo, and G. Rigau. A proposal for improving wordnet domains. In *LREC*, pages 3457–3462, 2012.

[62] B. C. Grau, Z. Dragisic, K. Eckert, J. Euzenat, A. Ferrara, R. Granada, V. Ivanova, E. Jiménez-Ruiz, A. O. Kempf, P. Lambrix, et al. Results of the ontology alignment evaluation initiative 2013. In *OM: Ontology Matching*, pages 61–100, 2013.

[63] N. S. C. C. Group. Glossary of terms - nhs jargon buster. https://www.sheffieldccg.nhs.uk/about-us/glossary-of-terms.htm, 2013. [Online; accessed 21-January-2020].

[64] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge acquisition*, 5(2):199–220, 1993.

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

[66] N. Guarino. Understanding, building and using ontologies. *International Journal of Human-Computer Studies*, 46(2-3):293–310, 1997.

[67] N. Guarino. Formal ontology in information systems. In *Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy*, volume 46, pages 3–15. IOS press, 1998.

[68] N. Guarino. Some ontological principles for designing upper level lexical resources. In *Proceedings of the 1st International Conference on Language Resources and Evaluation. Granada, Spain, 28-30 May*, 1998.

[69] N. Guarino and C. A. Welty. *An overview of OntoClean*. Springer, 2004.

[70] I. Gurevych, J. Eckle-Kohler, S. Hartmann, M. Matuschek, C. M. Meyer, and C. Wirth. Uby: A large-scale unified lexical-semantic resource based on lmf. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 580–590. Association for Computational Linguistics, 2012.

[71] A. Hajmoosaei and P. Skoric. Museum ontology-based metadata. In *2016 IEEE Tenth International Conference on Semantic Computing (ICSC)*, pages 100–103. IEEE, 2016.

[72] R. W. Hamming. Error detecting and error correcting codes. *The Bell system technical journal*, 29(2):147–160, 1950.

[73] I. Harrow, R. Balakrishnan, E. Jimenez-Ruiz, S. Jupp, J. Lomax, J. Reed, M. Romacker, C. Senger, A. Splendiani, J. Wilson, and P. Woollard. Ontology mapping for semantically enabled applications. *Drug discovery today*, pages 2068–2075, 2019.

[74] I. Harrow, E. Jiménez-Ruiz, A. Splendiani, M. Romacker, P. Woollard, S. Markel, Y. Alam-Faruque, M. Koch, J. Malone, and A. Waaler. Matching disease and phenotype ontologies in the ontology alignment evaluation initiative. *Journal of biomedical semantics*, 8(1):55, 2017.

[75] R. Huddleston and G. Pullum. The cambridge grammar of the english language. *Zeitschrift für Anglistik und Amerikanistik*, 53(2):193–194, 2005.

[76] I. G. Husein, S. Akbar, B. Sitohang, and F. N. Azizah. Review of ontology matching with background knowledge. In *2016 International Conference on Data and Software Engineering (ICoDSE)*, pages 1–6. IEEE, 2016.

[77] S. International. Snomed ct worldwide. https://www.snomed.org/snomed-ct/sct-worldwide, 2019. [Online; accessed 23-January-2020].

[78] S. H. Jihan and A. Segev. Humanitarian assistance ontology for emergency disaster response. *Intelligent Systems, IEEE*, 29(3):6–13, 2014.

[79] E. Jiménez-Ruiz and B. Cuenca Grau. Logmap: Logic-based and scalable ontology matching. In *International Semantic Web Conference*, pages 273–288. Springer, 2011.

[80] E. Jiménez-Ruiz, C. Meilicke, B. Cuenca Grau, and I. Horrocks. Evaluating mapping repair systems with large biomedical ontologies. *Description Logics*, 13:246–257, 2013.

[81] E. Jiménez-Ruiz, T. R. Payne, A. Solimando, and V. Tamma. Limiting logical violations in ontology alignnment through negotiation. In *Fifteenth International Conference on the Principles of Knowledge Representation and Reasoning*, pages 217–226.

[82] D. Jurafsky. *Speech & language processing*. Pearson Education India, 2000.

[83] A. Keselman, C. A. Smith, G. Divita, H. Kim, A. C. Browne, G. Leroy, and Q. Zeng-Treitler. Consumer health concepts that do not map to the UMLS: where do they fit? *Journal of the American Medical Informatics Association*, 15(4):496–505, 2008.

[84] S. G. Kolte and S. G. Bhirud. Word sense disambiguation using wordnet domains. In *2008 First International Conference on Emerging Trends in Engineering and Technology*, pages 1187–1191. IEEE, 2008.

[85] P. Lambrix and R. Kaliyaperumal. A session-based ontology alignment approach enabling user involvement 1. *Semantic Web*, 8(2):225–251, 2017.

[86] P. León-Araúz and P. Faber. Context and terminology in the multilingual semantic web. In *Towards the Multilingual Semantic Web*, pages 31–47. Springer.

[87] E. B. Lerner, D. V. Jehle, D. M. Janicke, and R. M. Moscati. Medical communication: do our patients understand? *The American journal of emergency medicine*, 18(7):764–766, 2000.

[88] D. Lewis, B. L. Chang, and C. P. Friedman. Consumer health informatics. In *Consumer Health Informatics*, pages 1–7. Springer, 2005.

[89] H. Li, Z. Dragisic, D. Faria, V. Ivanova, E. Jiménez-Ruiz, P. Lambrix, and C. Pesquita. User validation in ontology alignment: functional assessment and impact. *The Knowledge Engineering Review*, 34:e15, 2019.

[90] R. Lieber. Derivational morphology, 2017.

[91] E. J. Lowe. *More kinds of being: A further study of individuation, identity, and the logic of sortal terms.* John Wiley & Sons, 2015.

[92] A. Lyndon, M. G. Zlatnik, and R. M. Wachter. Effective physician-nurse communication: a patient safety essential for labor and delivery. *American journal of obstetrics and gynecology*, 205(2):91–96, 2011.

[93] B. Magnini and G. Cavaglia. Integrating subject field codes into wordnet. In *LREC*, 2000.

[94] B. Magnini, L. Serafini, and M. Speranza. Making explicit the semantics hidden in schema models. In *Proceedings of the Workshop on Human*

Language Technology for the Semantic Web and Web Services, held at ISWC-2003, Sanibel Island, Florida. Citeseer, 2003.

[95] J. Makhoul, F. Kubala, R. Schwartz, and R. Weischedel. Performance measures for information extraction. In *Proceedings of DARPA broadcast news workshop*, pages 249–252. Herndon, VA, 1999.

[96] F. Manola, E. Miller, and B. McBride. Rdf primer. *W3C recommendation*, 10(1-107):6, 2004.

[97] A. Marchetti, F. Ronzano, M. Tesconi, and M. Minutoli. Formalizing knowledge by ontologies: Owl and kif. *Relatório apresentado L'Istituto di Informatica e Telematica (IIT). Consiglio Nazionale delle Ricerche (CNR). Italia*, pages 2–14, 2008.

[98] R. Marinelli, M. Tiberi, and R. Bindi. Encoding terms from a scientific domain in a terminological database: Methodology and criteria. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, pages 2495–2500. European Language Resources Association (ELRA), 2008.

[99] B. Marr. How much data do we create every day? the mind-blowing stats everyone should read. https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-create-every-day-the-mind-blowing-stats-\everyone-should-read/#4a858f4d60ba, 2018. [Online; accessed 05-January-2020].

[100] F. Mata, P. Sánchez, I. Palomares, F. J. Quesada, and L. Martínez. Comas: A consensus multi-agent based system. In *2010 10th International Conference on Intelligent Systems Design and Applications*, pages 457–462. IEEE, 2010.

[101] J. P. McCrae, M. Arcan, K. Asooja, J. Gracia, P. Buitelaar, and P. Cimiano. Domain adaptation for ontology localization. *Web Semantics: Science, Services and Agents on the World Wide Web*, 36:23–31, 2016.

[102] D. L. McGuinness and F. Van Harmelen. Owl web ontology language overview. *W3C recommendation*, 10(10):2004, 2004.

[103] F. McNeill, D. Bental, P. Missier, J. Steyn, T. Komar, and J. Bryans. Communication in emergency management through data integration and trust: an introduction to the cem-dit system. In *Proceedings of the 16th International Conference on Information Systems for Crisis Response and Management*, pages 376–387, 2019.

[104] F. McNeill, P. Besana, J. Pane, and F. Giunchiglia. *Service Integration through Structure-Preserving Semantic Matching*, pages 64–82. IGI Global, 2009.

[105] F. McNeill, A. Gkaniatsou, and A. Bundy. Dynamic data sharing for facilitating communication during emergency responses. In *Proceedings of the 11th International Conference on Information Systems for Crisis Response and Management*, pages 369–373, 2014.

[106] R. Mizoguchi. Yamato: Yet another more advanced top-level ontology. In *Proceedings of the Sixth Australasian Ontology Workshop*, pages 1–16, 2010.

[107] A. Mol. *The body multiple: Ontology in medical practice*. Duke University Press, 2002.

[108] J. Moßgraber, F. Chaves, S. Middleton, Z. Zlatev, and R. Tao. The seven main challenges of an early warning system architecture. In *Proceedings of the 10th International ISCRAM. Conference*, pages 322–330, 2013.

[109] K. Nagatani, S. Kiribayashi, Y. Okada, K. Otake, K. Yoshida, S. Tadokoro, T. Nishimura, T. Yoshida, E. Koyanagi, and M. Fukushima. Emergency response to the nuclear accident at the fukushima daiichi nuclear power plants using mobile rescue robots. *Journal of Field Robotics*, 30(1):44–63, 2013.

[110] V. Nebot and R. Berlanga. Efficient retrieval of ontology fragments using an interval labeling scheme. *Information Sciences*, 179(24):4151–4173, 2009.

[111] H. Nefzi, M. Farah, I. R. Farah, and B. Solaiman. Towards a new ontology matching approach based on multi-criteria analysis methods. In *International Image Processing, Applications and Systems Conference*, pages 1–7. IEEE, 2014.

[112] C. P. M. Netten. *Machine learning for relevance of information in crisis response*. PhD thesis, FNWI, 2015.

[113] R. Occhiuzzi, F. Portilla, and P. Orefice. Creación de la extensión nacional uruguaya de snomed ct. In *VII Congreso Argentino de Informática en Salud (CAIS 2016)-JAIIO 45*, pages 73–78, 2016.

[114] P. Ochieng and S. Kyanda. Large-scale ontology matching: State-of-the-art analysis. *ACM Computing Surveys (CSUR)*, 51(4):75–110, 2018.

[115] G. of Canada. Canadian disaster database. https://cdd.publicsafety.gc.ca/srchpg-eng.aspx?dynamic=false, 2013. [Online; accessed 19-January-2020].

[116] W. H. Organization. *The ICD-10 classification of mental and behavioural disorders: clinical descriptions and diagnostic guidelines*. Geneva: World Health Organization, 1992.

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

[118] F. J. Quesada Real. Emergency Response extension for WordNet (100 terms from the UKCP lexicon), 3 2017.

[119] F. J. Quesada Real, F. McNeill, G. Bella, and A. Bundy. Improving dynamic information exchange in emergency response scenarios. In *Proceedings of 14th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2017)*, pages 824–833, 2017.

[120] F. J. Quesada Real, F. McNeill, G. Bella, and A. Bundy. Identifying semantic domains in emergency scenarios. In *15th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2018)*, pages 1130–1132, 2018.

[121] D. Robertson, F. Giunchiglia, S. Pavis, E. Turra, G. Bella, E. Elliot, A. Morris, M. Atkinson, G. McAllister, A. Manataki, P. Papapanagiotou, and M. Parsons. Healthcare data safe havens: towards a logical architecture and experiment automation. *The Journal of Engineering*, 2016(11):431–440, 2016.

[122] A. Roventini and R. Marinelli. Extending the italian wordnet with the specialized language of the maritime domain. In *Proceedings of the 2nd International Global Wordnet Conference*, pages 193–198, 2004.

[123] S. J. Russell and P. Norvig. *Artificial intelligence: a modern approach*. Pearson Education Limited, 3rd Edition, 2016.

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

[125] M. Sakurai. Communication platform for disaster response. In *International Conference on Design Science Research in Information System and Technology*, pages 223–227. Springer, 2016.

[126] D. M. Sánchez, J. M. Cavero, and E. M. Martínez. The road toward ontologies. In *Ontologies*, pages 3–20. Springer, 2007.

[127] A. Segev. Adaptive ontology use for crisis knowledge representation. *International Journal of Information Systems for Crisis Response and Management*, 1(2):16–30, 2009.

[128] M. Shamsfard, B. Helli, and S. Babalou. Omega: Ontology matching enhanced by genetic algorithm. In *2016 Second International Conference on Web Research (ICWR)*, pages 170–176. IEEE, 2016.

[129] P. Shvaiko and J. Euzenat. Ontology matching: state of the art and future challenges. *Knowledge and Data Engineering, IEEE Transactions on*, 25(1):158–176, 2013.

[130] K. Slabbekoorn, L. Hollink, and G. Houben. Domain-aware ontology matching. In *The Semantic Web–ISWC 2012*, pages 542–558. Springer, 2012.

[131] B. Smith. Ontology. In L. Floridi, editor, *In The Blackwell Guide to the Philosophy of Computing and Information*, pages 155–66. Oxford, UK: Blackwell Publishing Ltd, 2003.

[132] S. Staab and R. Studer. *Handbook on ontologies*. Springer Science & Business Media, 2010.

[133] G. Stoilos, G. Stamou, and S. Kollias. A string metric for ontology alignment. In *International Semantic Web Conference*, pages 624–637. Springer, 2005.

[134] V. C. Storey. Understanding semantic relationships. *The VLDB Journal*, 2(4):455–488, 1993.

[135] M. C. Suárez-Figueroa, A. Gómez-Pérez, and M. Fernández-López. The neon methodology for ontology engineering. In *Ontology engineering in a networked world*, pages 9–34. Springer, 2012.

[136] E. Thiéblin, O. Haemmerlé, N. Hernandez, and C. Trojahn. Survey on complex ontology matching. *Semantic Web*, (Preprint):1–39, 2019.

[137] T. Togia and F. McNeill. On-the-fly repair of structural and lexical ontology mismatches in multi-agent systems. *Artificial Intelligence Review*, 40(2):175–191, 2013.

[138] D. Truran, P. Saad, M. Zhang, and K. Innes. Snomed ct and its place in health information management practice. *Health Information Management Journal*, 39(2):37–39, 2010.

[139] M. Uschold and M. Gruninger. Ontologies and semantics for seamless connectivity. *ACM SIGMod Record*, 33(4):58–64, 2004.

[140] J. Vajda, E. Merrell, and B. Smith. Toward an ontology of commercial exchange. In *Proceedings of the Joint Ontology Workshops (JOWO), Graz*, 2019.

[141] S. Vázquez, A. Montoyo, and G. Rigau. Using relevant domains resource for word sense disambiguation. In *Proceedings of the International Conference on Artificial Intelligence, IC-AI '04, Volume 2 & Proceedings of the International Conference on Machine Learning; Models, Technologies & Applications, MLMTA '04*, pages 784–789. Citeseer, 2004.

[142] J. M. Vázquez Naya, M. Martínez Romero, J. Loureiro Pereira, C. R. Munteanu, and A. Sierra Pazos. Improving ontology alignment through genetic algorithms. In *Soft computing methods for practical environment solutions: Techniques and studies*, pages 240–259. IGI Global, 2010.

[143] J. Wang, Z. Ding, and C. Jiang. Gaom: Genetic algorithm based ontology matching. In *IEEE Asia-Pacific Conference on Services Computing (APSCC'06)*, pages 617–620. IEEE, 2006.

[144] T. Winograd. Understanding natural language. *Cognitive psychology*, 3(1):1–191, 1972.

[145] M. E. Winston, R. Chaffin, and D. Herrmann. A taxonomy of part-whole relations. *Cognitive science*, 11(4):417–444, 1987.

[146] X. Xue and J. Pan. An overview on evolutionary algorithm based ontology matching. *J. Inf. Hiding Multimed. Signal Process*, 9:75–88, 2018.

[147] X. Xue and Y. Wang. Using memetic algorithm for instance coreference resolution. *IEEE Transactions on Knowledge and Data Engineering*, 28(2):580–591, 2015.

[148] X. Xue, Y. Wang, and W. Hao. Optimizing ontology alignments by using nsga-ii. *International Arab Journal of Information Technology (IAJIT)*, 12(2), 2015.

[149] D. Yarowsky. One sense per collocation. In *Proceedings of the workshop on Human Language Technology*, pages 266–271. Association for Computational Linguistics, 1993.

[150] Q. T. Zeng and T. Tse. Exploring and developing consumer health vocabularies. *Journal of the American Medical Informatics Association*, 13(1):24–29, 2006.

[151] J. Zhang, J. Li, and Z. Liu. Multiple-resource and multiple-depot emergency response problem considering secondary disasters. *Expert Systems with Applications*, 39(12):11066–11071, 2012.

[152] Y. Zhao, J. Z. Pan, E. Thomas, N. Jekjantuk, and Y. Ren. Ontology languages and description logics. In *Ontology-Driven Software Development*, pages 51–67. Springer, 2013.