UNIVERSIDAD DE MURCIA

FACULTAD DE INFORMÁTICA

Methodology for the Enrichment of
Biomedical Knowledge Resources

Metodología para el Enriquecimiento de
Recursos de Conocimiento Biomédico

D. Manuel Quesada Martínez

2015

Methodology for the enrichment of biomedical knowledge resources

—

Metodología para el enriquecimiento de recursos de conocimiento biomédico

Authored by:

D. Manuel Quesada Martínez

Supervised by:

Dr. Jesualdo T. Fernández Breis

Dr. Robert D. Stevens

Tesis Doctoral - Doctorado en Informática

Dept. Informática y Sistemas

Universidad de Murcia

2015

This work has been funded by the *Ministerio de Economía y Competitividad*, *MINECO*, through the project TIN2010-21388-C02-02; and co-funded by the FEDER program and the Séneca Foundation through the project 15295/PI/10; and it has been possible thanks to the PhD fellowship of the National Program for the Training of Researchers, granted by the Spanish Ministry of Science and Innovation to Manuel Quesada Martínez (BES-2011-046192).

———

Este trabajo de investigación está financiado por el Ministerio de Economía y Competitividad, MINECO, mediante el proyecto TIN2010-21388-C02-02; y co-financiado por el Programa FEDER y la Fundación Séneca mediante el proyecto 15295/PI/10; y se ha podido realizar gracias a una beca de postgrado del Programa Nacional para la Formación de Personal Investigador, concedida por el Ministerio de Ciencia e Innovación a Manuel Quesada Martínez (BES-2011-046192).

Agradecimientos

A Jesualdo por su ayuda y guía a lo largo de este camino. Por ser un ejemplo de esfuerzo y trabajo diario, y por estar siempre dispuesto a ayudar. Gracias por darme la oportunidad de formar parte del grupo y aprender día a día con vosotros.

A Robert Stevens por sus comentarios y la co-dirección de este trabajo. Gracias por invitarme a realizar mi estancia más larga con el grupo que lidera. También agradecer a Nathalie Aussenac Gilles y Daniel Karlsson por haber dirigido mis otras dos visitas. Vosotros habéis hecho que mi tiempo allí haya merecido la pena. Sin duda, vuestra ayuda forma parte de esta tesis.

A los revisores que han contribuido con sus comentarios a mejorar las publicaciones aquí presentadas. A Paco, Marcos y otros profesores por darme la oportunidad de aprender a enseñar. A Domingo, Asun y Jose Juan por introducirme en el mundo de la investigación y por comprender mis decisiones.

A todos los compañeros de investigación que han compartido despacho todos estos años: desde España hasta México pasando por Colombia y otras partes del mundo. Vosotros habéis hecho que recuerde con cariño cada día. Extiendo el agradecimiento a todos aquellos que me han hecho sentir como en casa durante las temporadas en el extranjero; en especial a Eleni Mikroyannidi, Rafa Valencia y compañía.

A mi familia y amigos de los que siempre he recibido palabras de apoyo. A mis padres y hermana por transmitirme desde niño la tranquilidad necesaria para lograr este objetivo y por hacer que la vida sea mucho más sencilla. Gracias por todo. A mis cinco abuelos por cuidarme y permitirme tener una visión diferente del mundo.

Por último a María, quien se merece toda mi admiración y gratitud. Gracias por estar disponible en cada instante, por ser paciente y recibirme siempre con cariño, confianza, apoyo y comprensión. Gracias {at}.

Index of content

1. Introduction 1

2. State of the art 9

2.1. Ontologies . . . 9

2.1.1. Ontology components . . . 12

2.1.2. Knowledge Representation Languages . . . 14

2.1.3. RDF and RDF(S) . . . 17

2.1.4. Web Ontology Language (OWL) . . . 17

2.1.5. Reasoning and inferencing . . . 22

2.2. Biomedical knowledge resources . . . 23

2.2.1. Bioinformatics . . . 23

2.2.2. Clinical informatics and ontologies . . . 27

2.2.3. Biomedical knowledge resources . . . 27

2.3. Ontologies and identifiers in natural language . . . 29

2.3.1. Guidelines for naming identifiers . . . 32

2.3.2. Hidden semantics . . . 33

2.3.3. Lexically-suggest logically define . . . 35

2.4. Ontology engineering . . . 35

2.4.1. Methodologies for building ontologies . . . 35

2.4.2. Ontology enrichment . . . 39

2.4.3. Ontology matching . . . 44

2.4.4. Quality assurance in ontologies . . . 46

3. Objectives and methodology 47

3.1. Motivation . . . 47

3.2. Main research question 48
3.3. Objectives 49
3.4. Methodology 50

4. The OntoEnrich framework 53
4.1. General description of the method 53
4.2. Results and Applications 65
4.2.1. Lexical characterisation of biomedical ontologies 65
4.2.2. Lexical regularities and taxonomic relations 68
4.2.3. GO MF analysis 69
4.2.4. Gene Ontology cross-product extension 71
4.3. OntoEnrich tool 73

5. Publications composing the PhD Thesis 77
5.1. Lex. Characteriz. of Bio-ontologies by the Inspection of Reg. in Label . . 77
5.2. Prioritising LPs to Increase Axiomatisation in Biomedical Ontologies . . 91
5.3. Appr. the axiomatic enrichment of the GO from a lex. perspective 101

6. Conclusions and future work 117
6.1. Contributions 117
6.2. Research questions 118
6.3. Brief discussion of future work 120
6.4. Publications and contribution in Conferences 123
6.4.1. Publications JCR 123
6.4.2. International Conferences and workshops 124

7. Resumen 127
7.1. Objetivos 130
7.2. Metodología 131
7.3. Resultados y conclusiones 131

Bibliography 153

A. Complexity analysis of the graph 155

Chapter 1

Introduction

The Semantic Web [Tim, Lee et al., 2001, Shadbolt et al., 2006] is the extension of the World Wide Web that enables people to share content beyond the boundaries of applications and websites[1]. Semantic Web technologies enable people to create data stores on the Web, build vocabularies, and write rules for handling data. Semantic Web technologies have been applied in the modelling of life science leading to the Life Sciences Semantic Web [Good and Wilkinson, 2006].

Ontologies are considered one of the pillars of the Semantic Web[2]. Although we will further explain what an ontology is in section 2, in brief, an ontology is a set of logical axioms that are designed to account for the intended meaning of a vocabulary [Guarino, 1998]; in other words, it is a representation that captures the categories of objects in a field of interest and the relationships that those objects have to each other in such a way that it is possible to recognise category membership. The Gene Ontology (GO) [Ashburner et al., 2000] is possibly the most prominent example of the success of ontologies in bioinformatics. The GO project is a collaborative effort to address the need for consistent descriptions of gene products across databases. But the GO is not unique. For example, in the medical side, SNOMED CT [Cornet and de Keizer, 2008] is a clinically validated and semantically rich controlled vocabulary[3]. One of its aims is to enable consistent, processable representation of clinical content in electronic health records, and it is already used in more than 50 countries. Alt-

[1] http://semanticweb.org/

[2] http://semanticweb.org/wiki/Ontology.html

[3] http://www.ihtsdo.org/snomed-ct/what-is-snomed-ct

hough it is originally released as tab-delimited text files that represent the components of SNOMED CT, these files can be converted into an ontology using an automatic process [The International Health Terminology Standards Development Organisation, 2015].

The ontology content is represented using three types of components: (1) concepts, (2) descriptions and (3) relationships. Figure 1.1 shows the general design of an ontology, using the Gene Ontology as example. Concepts represent "things" in reality and they are represented as pink boxes in Figure 1.1 like `'binding'`, `'protease binding'` or `'protein binding'`. Moreover, concepts are represented in a hierarchical manner. Using hierarchical relationships (`isA` in the figure) more specific concepts can be defined. For example, it can be seen that `'protein binding'` is a type of `'binding'`. `IsA` stands for the type of relation that means that a concept is a sub type of another concept, but many more could exist (see `'regulates'` in Figure 1.1).

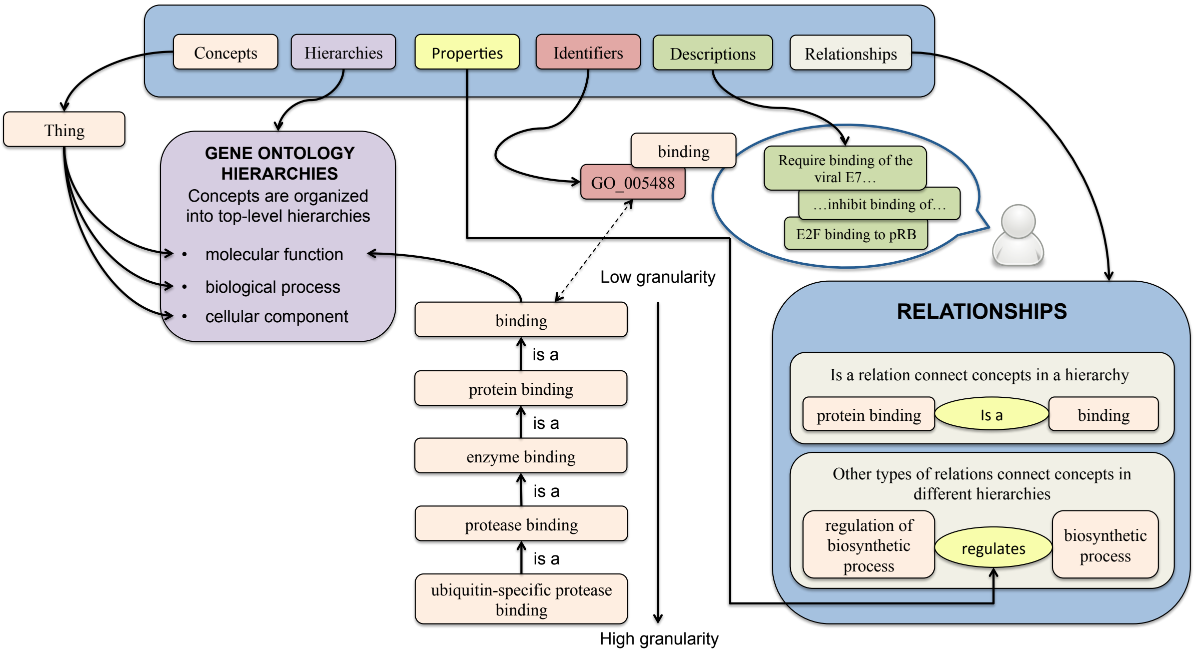


Figure 1.1: Gene Ontology design

Often, the first levels of the hierarchy can be used to represent different subdomains of knowledge. For example, in Figure 1.1 this is shown as purple boxes. The GO project has developed three structured knowledge branches that describe the biological processes, cellular components and molecular functions associated with gene products in a species-independent manner[4]. Knowledge representation languages provide a formalism based on logical axioms that enables machines to process content of the ontology and infer that `‘protease binding’` is a type of `‘binding’` despite there being no direct relation between them. The use of ontologies has several benefits. Those applications that use ontologies to formalise their domain could take advantage of reasoning processes. Moreover, each concept has a unique identifier, so applications that describe their domain using ontology concepts instead of describing it in natural language would avoid imprecisions, and the semantic interoperability of biological data would be closer.

In the last 15 years, the biomedical research community has increased its effort in the development of ontologies used to represent biomedical knowledge and there is no reason to expect this to change in the future [Hoehndorf et al., 2014]. As a consequence of their success, biomedical ontologies are usually built in community with a high level of activity. Then an ontology is the result of a collaborative work between different experts [Malone and Stevens, 2013]. We point out two profiles of experts:

- *Domain experts*: have further knowledge of the domain to represent, but they might not have enough ontological background to codify it properly in the ontology.
- *Ontology developers*: their knowledge about the domain is limited. They focus on formal aspects of the ontology and whether the knowledge within the ontology properly represents the domain.

The ontology development teams usually have members of those two profiles. In Figure 1.1 (top right) we can observe a domain expert. This expert interprets the domain in terms of descriptions, while an ontology developer contextualises such descriptions according to the concepts and relationships. This thesis aims at contributing to the enrichment of ontologies built by domain experts, which are rich in the knowledge about the domain but low in the axiomatisation that make this knowledge up to be processed by computers.

Moreover, having more and larger ontologies makes the maintenance of ontologies a difficult task due to collateral effects of individual changes. The types of changes made include new concepts, new descriptions, new relationships between concepts, as well as updates and retirement/deprecation of any of these components. However, the larger

[4] `http://geneontology.org/page/documentation`

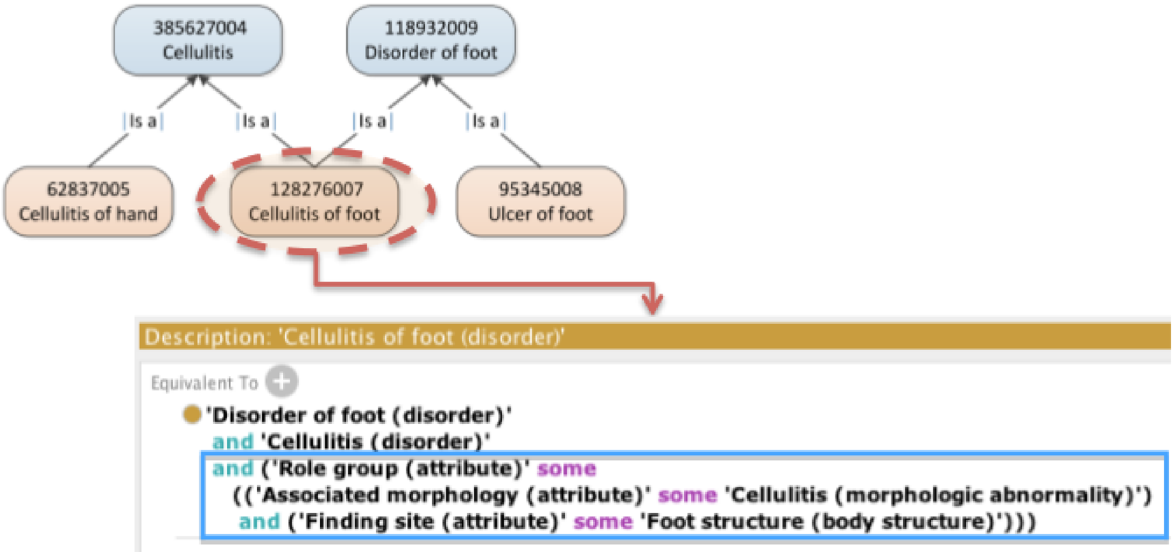
and more complex the ontology is, the more difficult it is to evaluate side effects that it might have over other ontology components. Also the creation of a new concept and its contextualization within the ontology can be a problem due to the formalism and complexity of ontologies, which goes beyond being understood as hierarchies of concepts easily understood by both profiles.

Let us follow a simplified example to explain such a complexity. In this case, we use the SNOMED CT domain. A general practitioner (*domain expert*) is interested in representing a type of cellulitis that has occurred in a part of the foot; cellulitis is a common skin infection caused by bacteria[5]. At first glance, this new concept is a kind of disorder. So inspecting the disorder hierarchy in SNOMED CT, the domain expert decides to define this new term, named `'Cellulitis of foot'`, as a type of `'Cellulitis'` and a type of `'Disorder of foot'`. The graphical representation of this relation can be seen in the upper part of Figure 1.2. However, an *ontology developer* inspects the term and decides to add attributes that logically complete the definition linking it with other concepts representing for example body structures. In the bottom box the formal definition is shown. In the "Equivalent To" section, some axioms that represent necessary and sufficient conditions of a concept to remain to the class that represent the disorder `'Cellulitis of foot'` are defined. The previous example could be harder given the real size of biomedical ontologies. For example, SNOMED CT (version 2015AA released in BioPortal on 06/09/2015) has around 316 031 concepts and GO (version released in BioPortal on 22/09/2015) 43 716, to this numbers the logical relations between these concepts should be added. Another aspect that increases the complexity of the maintenance of biomedical ontologies is their constant change (e.g. SNOMED CT has been released twice a year since 2002[6]). Although ontology editors usually provide some guidelines that help in the construction of ontologies, the development of domain independent methods and tools that contribute to the maintenance and quality assurance in ontologies is important [Rogers, 2006].

Quality assurance methods are still a challenge to which this thesis wishes to contribute by exploiting the *hidden semantics* codified in ontology identifiers. Quality assurance methods have been applied for different purposes and ontologies [Ceusters et al., 2004, Rogers, 2006, Ceusters, 2006, De Coronado et al., 2009,

[5] `https://www.nlm.nih.gov/medlineplus/ency/article/000855.htm`

[6] `http://ihtsdo.org/fileadmin/user_upload/doc/en_us/tig.html?t=rf2_title`



Description: 'Cellulitis of foot (disorder)'

Equivalent To

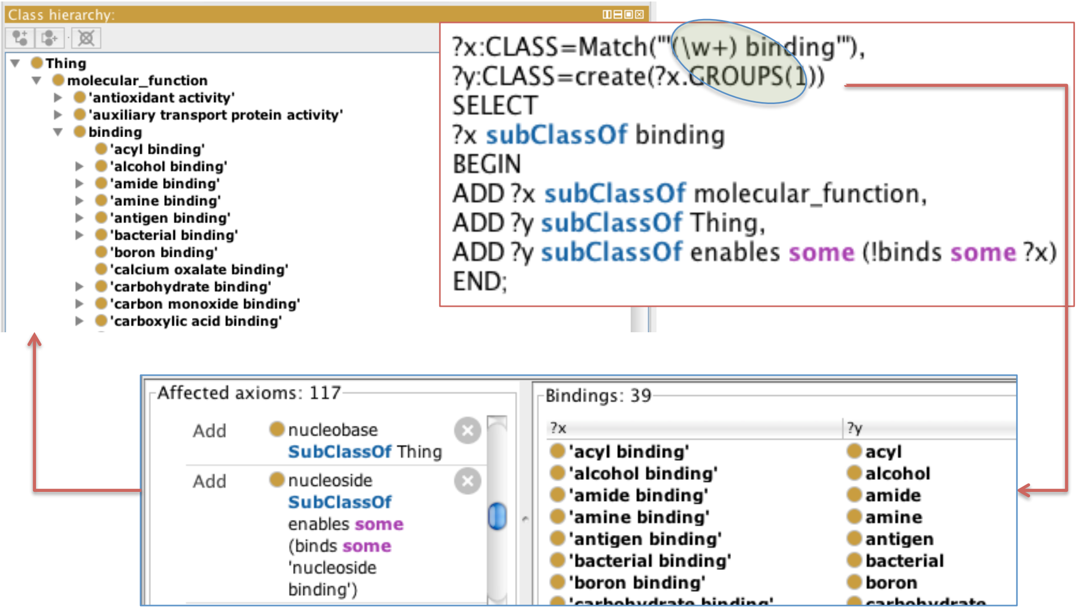
```
● 'Disorder of foot (disorder)'
  and 'Cellulitis (disorder)'
  and ('Role group (attribute)' some
    (('Associated morphology (attribute)' some 'Cellulitis (morphologic abnormality)')
    and ('Finding site (attribute)' some 'Foot structure (body structure)')))
```

Figure 1.2: Example of the logical representation of "Cellulitis of foot (disorder)" in SNOMED CT.

Verspoor et al., 2009, Mikroyannidi et al., 2011, Rector and Iannone, 2012]. The starting point for this work is [Fernandez-Breis et al., 2010], which describes the process of taking an axiomatically lean ontology and enriching it creating new formal relationships. A description of the process followed is:

1. Inspect the ontology to find out what needs to be revealed as axioms.
2. Develop patterns of axioms based on natural language definitions of concepts.
3. Identify supporting ontologies or modules that capture entities within the developed patterns.
4. Apply patterns across the source ontology that transform implicit information codified in natural language in explicit information codified as axioms.
5. Run a reasoner and inspect the resulting ontology.

The method was applied to enrich the GO Molecular Function Ontology. Figure 1.3 shows as example of the pattern *"X binding"*. This pattern is based on the definition of



```
Thing
  molecular_function
    'antioxidant activity'
    'auxiliary transport protein activity'
    binding
      'acyl binding'
      'alcohol binding'
      'amide binding'
      'amine binding'
      'antigen binding'
      'bacterial binding'
      'boron binding'
      'calcium oxalate binding'
      'carbohydrate binding'
      'carbon monoxide binding'
      'carboxylic acid binding'
```

```
?x:CLASS=Match("'(\w+) binding'"),
?y:CLASS=create(?x.GROUPS(1))
SELECT
?x subClassOf binding
BEGIN
ADD ?x subClassOf molecular_function,
ADD ?y subClassOf Thing,
ADD ?y subClassOf enables some (!binds some ?x)
END;
```

Affected axioms: 117

```
Add  nucleobase
     SubClassOf Thing
Add  nucleoside
     SubClassOf
     enables some
     (binds some
     'nucleoside
     binding')
```

Bindings: 39

?x	?y
'acyl binding'	acyl
'alcohol binding'	alcohol
'amide binding'	amide
'amine binding'	amine
'antigen binding'	antigen
'bacterial binding'	bacterial
'boron binding'	boron
'carbohydrate binding'	carbohydrate

Figure 1.3: This example shows how to define a patter (top right part) that can be used to systematically enrich the source ontology (top left part shows a piece of the GO-MF hierarchy). This pattern, "X binding", analysis labels that end with the word "binding". For this example, there are 39 cases that follow this pattern so 117 logical axioms will be created (bottom right).

the concept `'binding'` as the selective, non-covalent, often stoichiometric, interaction of a molecule with one or more specific sites on another molecule. This description defines a *knowledge pattern*, and based on this *knowledge pattern* the hierarchy of concepts is manually inspected with the goal of finding a regular structure within the labels that lets us convert the *knowledge pattern* into an OPPL pattern[7] (see Figure 1.3 top right). The OPPL pattern creates axioms by dissecting information codified in natural language; OPPL is an abstract formalism that allows for manipulating ontologies written in OWL. The exploration of the hierarchy of concepts reveals that the molecule that binds the binding is codified in natural language in the subtypes of binding. For example, `'alcohol binding'` is a specific type of `'binding'` enabled by an `'alcohol'` molecule, and this information is not logically codified in the original ontology.

The enrichment performed in [Fernandez-Breis et al., 2010] demonstrated the interest

[7] `http://oppl2.sourceforge.net/`

and benefits of this kind of enrichment in biomedical ontologies. However, the application of the process was tedious as the identification of knowledge and axiomatic patterns was performed manually, so automatic methods would be desirable. If so, an automatic method that performed the analysis of textual content within ontologies to elucidate *hidden semantics* could be systematically applied to other and even new versions of the same ontology, so that this contributes to the quality assurance of ontologies; this is the main motivation of this thesis. Such a generic approach would provide new insights into the engineering of biomedical ontologies and can contribute to guide ontology developers in the enrichment of *biomedical knowledge resources*. In this work, *biomedical knowledge resources* are ontology repositories like BioPortal, which in September 2015 contained 478 ontologies and controlled vocabularies.

The publications composing the PhD Thesis can be found in section 5 and are now presented:

- *Lexical characterization of Bio-Ontologies by the inspection of regularities in labels:*

  Hundreds of biomedical ontologies have been produced, with many of the significant, widely used ones being developed in collaborative efforts and following a set of construction principles, which include using a systematic naming convention for their labels. Despite their success, many of these ontologies lack of a rich axiomatisation that would expose the wealth of knowledge in the ontologies to computational reasoning. Previous work suggests that exploiting the structure of the labels may contribute to an axiomatic enrichment. Hence, in this work we perform a study of the structure of the labels of the ontologies available in BioPortal to classify them in terms of potential interest for axiomatic enrichment.

- *Prioritizing lexical patterns to increase axiomatisation in biomedical ontologies:*

  The aim of this work is to identify which *lexical regularities* are more promising for ontology enrichment. For this, we propose metrics for suggesting which *lexical regularities* should be the starting point to enrich complex ontologies. Our method determines the relevance of *lexical regularities* by measuring its locality in the ontology, that is, the distance between the classes associated with the regularity, and the distribution of them in a certain module of the ontology. The methods have been applied to four significant biomedical ontologies including the Gene Ontology and SNOMED CT.

The metrics provide information about the engineering of the ontologies and the distribution of classes that exhibit *lexical regularities*. Our method enables the suggestion of links between classes that are not made explicit in the ontology. We propose a prioritisation of the lexical patterns found in the analysed ontologies. Developers migth use this information to improve the axiomatisation of their ontologies.

- *Approaching the axiomatic enrichment of the Gene Ontology from a lexical perspective:*

  The main goal of this work is to extend our method with the cross-products extension (CPE) metric, which estimates the potential interest of a specific regularity for reconstructing cross-products. Cross-product extensions of GO have been recently used by the GO consortium to enrich this ontology. Cross-products are generated by establishing axioms that relate a given GO class to classes from the GO or other biomedical ontologies.

  The results obtained in this study show that GO class labels are highly regular in lexical terms, and the exact matches with labels of external ontologies affect 80 % of the GO classes. The CPE metric reveals that 31.48 % of the classes that exhibit regularities have fragments that are classes into two external ontologies that are selected for our experiment, namely, the Cell Ontology [Bard et al., 2005] and the Chemical Entities of Biological Interest ontology [Degtyarenko et al., 2007]. Moreover, 18.90 % are fully decomposable into smaller parts. Our results show that the CPE metric permits our method to detect GO cross-product extensions with a mean recall of 62 % and a mean precision of 28 %. The study is completed with an analysis of false positives to explain this precision value.

These results support the claim that the lexical approach can contribute to the axiomatic enrichment of biomedical ontologies and that it can provide new insights into the engineering of biomedical ontologies. These three publications represent a scientific unit as all of them contribute to develop the method presented in this thesis.

Chapter 2

State of the art

2.1. Ontologies

Throughtout history, the word "ontology" has been used by many authors from different backgrounds. As a consequence different interpretations of its meaning can be found in the literature. In Philosophy, Aristoteles (384- 322 BC) was one of the first in using ontologies in his attempt *"to classify the things in the world, where it is employed to describe the existence of the beings in the world"* [Studer et al., 1998]. After this first definition, many authors in Philosophy have taken and interpreted this term being ontologies a branch of philosophy [Smith, 2009], and ontologies still have an important role in modern Philosophy[1].

In 1991, Neches et al. explained the necessity of enabling sharing and reuse of knowledge bodies in a computational form [Neches et al., 1991, Studer et al., 1998] for developing large and more complex knowledge-based systems [Gonzalez and Dankel, 1993]. According to Neches's definition, *"an ontology defines the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary."* This was the first attempt to use of ontologies in the field of Artificial Intelligence (AI). Neches's definition resembles Quine's ontology philosophical interpretation: what exists is what can be quantified over [Fernández-Breis, 2003]. Then for AI systems, what "exists" is that which can be represented.

[1] `http://www.ontology.co/`

Two years later Gruber, who was a coauthor in [Neches et al., 1991], authored one of the most referenced definitions of ontology in AI, having received 14 286 citations in Google Scholar as of September 2015 [Gruber, 1993]:

> *"An ontology is an explicit specification of a conceptualisation. The term is borrowed from philosophy, where an ontology is a systematic account of Existence."*

We highlight three keywords in Gruber's definition: explicit, specification and conceptualisation. Explicit means clear and exact[2]. A specification is a detailed description of how something should be done, made, and so on[3]. Finally a conceptualisation is the form of an idea or principle in your mind[4]. Figure 2.1 shows an extract of Aristotle´s classification of animals (taken from[5]). Two ontologies could be different in the vocabulary used (using cat or the spanish word *gato*, for instance) while sharing the same conceptualisation.

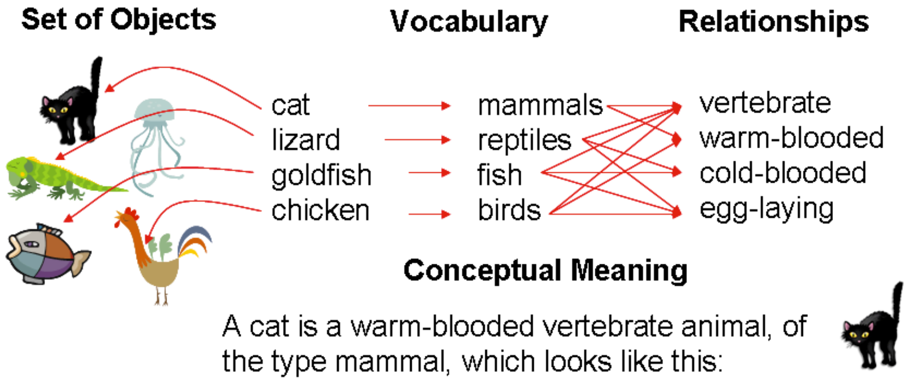


Figure 2.1: Ontology as an explicit specification of a conceptualisation.

However, not all the ontologies that satisfy Gruber´s definition are useful for sharing or machine consuming. For example, Figure 2.1 shows an ontology because it encodes knowledge about a domain, but it is far from being codified in a machine-processable form. In this sense, Borst clarifies Gurber's definition by replacing the word "explicit"

[2] http://dictionary.cambridge.org/es/diccionario/ingles/explicit

[3] http://dictionary.cambridge.org/es/diccionario/ingles/specification

[4] http://dictionary.cambridge.org/es/diccionario/ingles/conceptualize

[5] ftp://ftp.ebi.ac.uk/pub/databases/chebi/tutorial/chebi_tutorial_block3.doc

by “formal” and adding “shared”: *“ontologies are formal specifications of shared conceptualisations”* [Borst, 1997].

In [Guarino, 1995] a discussion about some formal ontological distinctions is addressed. In [Guarino, 1998], the author went beyond Borst’s clarification by discussing the use of the word *conceptualisation* in Gruber´s definition. According to Guarino, a *conceptualisation* refers to the ordinary mathematical extensional definition. For example, the extensional definition of mammals would consist on listing all that entities in reality that remain to that category. For example, in Figure 1.3 an `‘acyl binding’` is a type of `‘binding’` so everything that is in the category of `‘acyl binding’` is also a `‘binding’`. However, ontologies as engineering artifacts use intentional definitions, which gives the meaning of a category by specifying the necessary and sufficient conditions for belonging to such a category. While extensional relations reflect a particular state of affairs, intensional relations, called *conceptual relations* are focused on the meaning of these relations. As a result of this discussion, Guarino defines an ontology as follows [Guarino, 1998]:

> *“In the philosophical sense, we may refer to an ontology as a particular system of categories accounting for a certain vision of the world. As such, this system does not depend on a particular language: Aristotle´s ontology is always the same, independently of the language used to describe it. On the other hand, in its most prevalent use in AI, an ontology refers to an engineering artifact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary words. This set of assumptions has usually the form of a first-order logical theory, where vocabulary words appear as unary or binary predicate names, respectively called concepts and relations. In the simplest case, an ontology describes a hierarchy of concepts related by subsumption relationships; in more sophisticated cases, suitable axioms are added in order to express other relationships between concepts and to constrain their intended interpretation.”*

In conclusion, nowadays there is no agreement in how to define ontologies as computational artifacts. On the one hand, *“researchers seem to have been much more interested in the nature of reasoning rather than in the nature of the real world”* [Guarino, 1995].

As a consequence of this, *"the computer science view of ontology is somewhat narrower, where an ontology is the working model of entities and interactions either generically or in some particular domain of knowledge or practice"* [Stevens et al., 2000]. On the other hand, in a strict philosophical sense, *"... ontological engineering aims not for truth, but rather, merely, for adequacy to whatever is the pertinent application domain"* [Smith, 2003], which were discussed in the origin as the *"essential ontological promiscutiy of AI"* [Genesereth and Nilsson, 1987]. Other authors provide a definition based on the approach they take to build their ontologies. Different definitions provide different and complementary points of view of the same reality. In this thesis we adopt Guarino´s interpretation as it is the one adopted in *biomedical knowledge repositories* (further explained in section 2.2).

2.1.1. Ontology components

Regardless of the definition of ontology adopted, knowledge of ontologies is formalised using different kinds of elements. Although the ontology languages chosen for codifying the ontology will allow one to define different types of elements, in general there are 4 main types that form the core of all ontologies: concepts, individuals, relations and axioms.

- **Concepts:** a concept represents set of classes, entities or "things" within a domain. They provide the abstraction mechanism for grouping resources with similar characteristics. A concept, used in a broad sense, can be anything about which something is said, it refers to what is general in reality. Concepts are also called terms, classes, universals, types or kinds.
- **Individuals:** they are used to represent concrete elements that pertain to a certain domain, which is described in terms of concepts. They are things that the ontology describes or potentially could describe. Individuals are also called instances or particulars.

Some authors claim that "concepts" refer to what is general in reality. Instances refers to what is particular in reality; entities (including processes) that exist in space and time and stand to each other in a variety of instance-level relations [Smith, 2004, Smith et al., 2005]. However, deciding whether something is a concept or an instance is difficult and often depends on the application. For example, `‘Atom’` is a concept and `‘Potassium’` is an instance of that concept. It could be argued that `‘Potassium’` is a concept representing the different instances of potassium and its isotopes, and so on. This is a well known and open question in knowledge management research [Stevens et al., 2000].

- **Relations:** they describe interactions between concepts of the domain. Relations can be expressed directly between individuals, between concepts or both. Relations can have different nature and (logical) properties.
- **Axioms:** they model statements that model sentences that are always true so they are used to constrain values for classes or instances. In this sense the properties of relations are kinds of axioms.

Let us now explain relations and their properties using the taxonomical relation of hypernymy `isA`. This relation is used to indicate that a concept C_1 is a subtype of C_2, and it has the property inverse with the relation hyponym. For example, `‘Mitochondrion’ isA ‘Intracellular Organelle’`, and `‘Intracellular Organelle’ isA ‘Organelle’`. Moreover, another property of the hypernymy relation is its transitivity. This is to say, if C_1 is a type of C_2 and C_2 is a type of C_3, it can be inferred that C_1 is a type of C_3, so it could be inferred that `‘Mitochondrion’ isA ‘Organelle’`. Then, automated reasoning techniques allow a computer system to draw conclusions from the knowledge represented in a machine in a interpretable form [Stephan et al., 2007]. Apart from hypernymy-hyponym relations, another type of taxonomical relations are holonym-meronym, which define a possessive hierarchy. For example, `‘Mitochondrion’ partOf ‘Cytoplasm’`, and `‘Cytoplasm’ partOf ‘Cell’`.

The 4 previous taxonomical relations let ontologies be organised in a hierarchical manner. All those relations that are not taxonomical are considered associative relations, which are used to relate concepts across hierarchies. For example, an associative relation can represent: (1) the function of a concept in `‘Protein’ hasFunction ‘Receptor’`, (2) locative relationships in `‘Chromosome’ hasSubcellularLocation ‘Nucleus’` and so on. Other types of relations and a further descriptions of its properties can be found in [Gómez-Pérez, 1999] and [Fernández-Breis, 2003]. Finally, it should be pointed out that

Component/Author		Gruber, 1993	Guarino, 1995	Gomez, 1999	Stevens, 2000	Smith, 2004	Fernández-Breis, 2003	Lord, 2010	Pesquita, 2012	Others
Class	Term									X
	Concept		X	X	X	-		X	X	
	Class	X					X	X	X	
	Universal					X		X		
	Type							X		
Relation	Relation	X	X	X	X	X	X	X	X	
	Function	X					X			
	Axiom	X	X	X	X		X			
Instances	Instance	X		X	X		X	X	X	
	Individual							X		
	Particular					X		X		

Table 2.1: Ontology components and terminology used by different authors

the terminology used to refer to ontologies components can vary from one author to other, despite the meaning is similar. The previous definitions about ontology components were built as a study of the work shown in Table 2.1. In such a table we show the terminology used by different authors, and group them according to the 4 main ontology components described before.

2.1.2. Knowledge Representation Languages

In the previous section, we claimed that the ontology components are also influenced by the knowledge representation language (KRL) chosen. In this section we explain different KRLs. According to the definition of Guarino: *"... an ontology refers to an engineering artifact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the indented meaning of the of the vocabulary words. This set of assumptions has usually the form of a first-order logical theory..."*. This definition explicitly references to first-order (predicate) logic (FOL) as the KRL for representing ontologies as engineering artifacts. However, other KRLs such as semantic networks and frames had been previously used.

- A semantic network is a graph whose nodes represent concepts and whose arcs represent relations between these concepts. Frame systems and semantic networks can be identical in their expressiveness but use different representation. While the semantic network is that of a graph with concept nodes linked by relation arcs, the frame draws concepts as boxes, i.e. frames, and relations as slots inside frames that can be filled by other frames. [Stephan et al., 2007].

FOL differs from its predecessors in that they are equipped with formal, logic-based semantics [Baader et al., 2008]. Then, the expressivity of KRLs used in semantic networks and frames is lower than FOL. For this reason, FOL is the prevalent and single most important knowledge representation formalism.

FOL provides a notion of *logical consequence* and *universal truth* that can be described in terms of *model-theoretic semantics.* This formalism enables a process called *deduction* or *inferencing* previously mentioned. Using *inferencing*, computer systems can base decisions on reasoning about domain knowledge similar to humans. This process is supported by algorithms for deduction, which are required to be sound and complete. However, in general, FOL *inferencing* algorithms are *semi-decidable.* This means that given a theory and a query statement, to terminate with positive answer in finite time is possible whenever the statement is a logical consequence of the theory. On the contrary, if the statement is not a logical consequences of the theory the termination is not required, and indeed, termination (with the correct negative answer) cannot be guaranteed in general [Stephan et al., 2007]. Semi-decidablity is a problem in the representation of ontologies. Ontologies as engineering artifacts that are used by computers for supporting users in different tasks. For this reason, the decidability of the KRL is important as users hope to have answers to their queries.

Figure 2.2 attempts to give an overview of the most important KRLs for representing ontologies in the Semantic Web [Stephan et al., 2007]. On the left, KRLs based on FOL are shown. On the right other languages based on other paradigms are shown. We focus our attention in KRLs based on FOL. Although FOL is undecidable, there are subsets of it, called Description Logics (DL), which just contains essential decidable fragments [Baader, 2003]. Figure 2.2 distinguishes between undecidable and decidable languages with a horizontal line. KRLs based on DL are considered to have a level of expressivity that is proper for representing biomedical knowledge.

Expressivity of DL languages

The expressivity of logic is encoded in the labels of starting from the basic logic. In [Schmidt-Schau and Smolka, 1991] the fist naming scheme for DLs was proposed: starting from a basic DL $\mathcal{AL}$, the addition of a constructors is indicate by appending a corresponding letter; e.g., $\mathcal{ALC}$ is obtained from $\mathcal{AL}$ by adding the complement operator

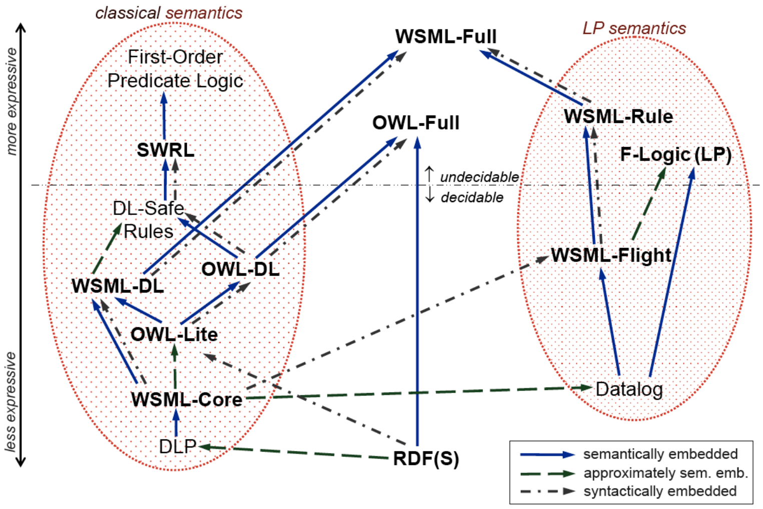
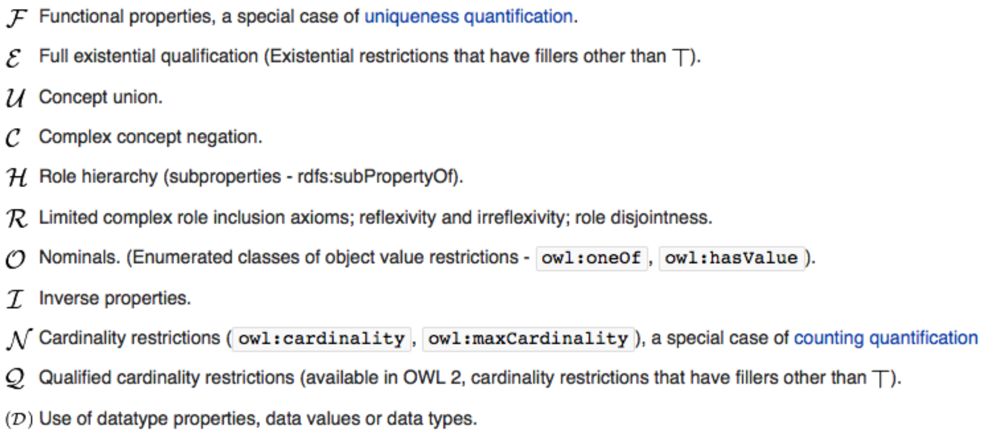


Figure 2.2: An overview of Semantic Web languages.

(¬) [Baader et al., 2008]. The basic logic $\mathcal{AL}$ allows: (1) atomic negation, concept intersection, universal restrictions and limited existential quantification[6]. Figure 2.3 shows a summary of the extensions for the basic DL logic. The expressive power of a language like OWL is determined by the class (and property) constructors supported, and by the kinds of axioms that can occur in an ontology [Horrocks et al., 2003].



$\mathcal{F}$ Functional properties, a special case of uniqueness quantification.

$\mathcal{E}$ Full existential qualification (Existential restrictions that have fillers other than $\top$).

$\mathcal{U}$ Concept union.

$\mathcal{C}$ Complex concept negation.

$\mathcal{H}$ Role hierarchy (subproperties - rdfs:subPropertyOf).

$\mathcal{R}$ Limited complex role inclusion axioms; reflexivity and irreflexivity; role disjointness.

$\mathcal{O}$ Nominals. (Enumerated classes of object value restrictions - `owl:oneOf` , `owl:hasValue`).

$\mathcal{I}$ Inverse properties.

$\mathcal{N}$ Cardinality restrictions (`owl:cardinality` , `owl:maxCardinality`), a special case of counting quantification

$\mathcal{Q}$ Qualified cardinality restrictions (available in OWL 2, cardinality restrictions that have fillers other than $\top$).

($\mathcal{D}$) Use of datatype properties, data values or data types.

Figure 2.3: Extensions of the basic DL

[6] https://en.wikipedia.org/wiki/Description_logic

A DL knowledge base is made up of two parts: a terminological part (called the TBox) and an assertional part (called the ABox), each part consisting of a set of axioms. On the one hand, the TBox is the scheme and it contains concepts, properties and restrictions. On the other hand, the ABox contains individuals. In DL the fundamental modelling concept is the axiom. Axioms are translated to first-order predicate statements [Baader et al., 2008].

2.1.3. RDF and RDF(S)

RDF (Resource Description Framework) is a web standard that represents data using triplets. Each triplet is composed by a subject, predicate and object. The predicate express the nature of the relation and the components are identified though a Unique Resource Identifier (URI). The RDF scheme (RDFS) is an extension of RDF vocabulary that includes semantics; users can define classes, organise them by hierarchies, define relations and set domain and ranges. The RDF(S) language can be seen in the bottom part of Figure 2.2.

> *Example:* Using RDFS users can declare classes like `Country`, `Person`, `Student` and `Canadian`. Using RDFS users can state that `Canada` and `England` are both instances of the class `Country`.

2.1.4. Web Ontology Language (OWL)

The World Wide Web Consortium (W3C) formed the Web Ontology Working Group, whose goal was to develop an expressive language suitable for application in the Semantic Web. The result of this endeavor was the Web Ontology Language (OWL), which became a W3C recommendation in February 2004[7].

Predecessors of OWL

One of the first attempts at defining an ontology language for deployment on the Web was RDF(S). Another attempt was SHOE, which is a frame-based language with an XML syntax that could be safely embedded in existing HTML documents. SHOE also uses

[7] `http://www.w3.org/TR/owl-features/`

URI reference for names. A new language called DAML-ONT was therefore developed that extended RDF with language constructors from object-oriented and frame-based knowledge representation language. However, the lack of formality of RDF and RDFS specification soon led to arguments about the meaning of language constructs, such as the domain and range.

Almost in parallel to the development of DAML-ONT, OIL was developed being the first ontology language that combines elements from DL, frame languages and web standards such as XML and RDF. It became obvious to both the DAML-ONT and OIL group that their objectives could best be served by combining their efforts, the result being the merging of DAML-ONT and OIL to produce DAML+OIL. The DL derived language constructors of OIL were retained in DAML+OIL, but the frame structure was largely discarded in favour of DL style axioms, which were more easily integrated with RDF syntax. Given that OWL[8] was not the first web-enable ontology language. OWL had to maintain as much compatibility as possible with other DL based ontology existing languages, including SHOE, OIL, DAML+OIL, and so on [Horrocks et al., 2003].

OWL expressivity and profiles

OWL defines, in turn, different profiles with different expressivities. From less to more expressivity: OWL-Lite, OWL-DL and OWL-Full[9]. In Figure 2.2 their relation and contextualisation in term of expressively and decidability is shown. As it is claimed in the OWL specification document:

- OWL Full and OWL DL support the same set of OWL language constructs. Their difference lies in the restrictions on the use of some of those features and on the use of RDF features. OWL Full allows free mixing of OWL with RDF Schema and, like RDF Schema. It does not enforce a strict separation of classes, properties, individuals and data values. OWL DL puts constraints on the mixing with RDF and requires disjointness of classes, properties, individuals and data values. These constraints make it decidable in contrast with OWL-Full.

- OWL Lite is a sublanguage of OWL DL that supports only a subset of the OWL language constructs. OWL Lite is particularly targeted at tool builders, who want

[8] `http://www.w3.org/TR/owl-ref/`

[9] `http://www.w3.org/TR/owl-ref/#Sublanguages`

to support OWL, but want to start with a relatively simple basic set of language features.

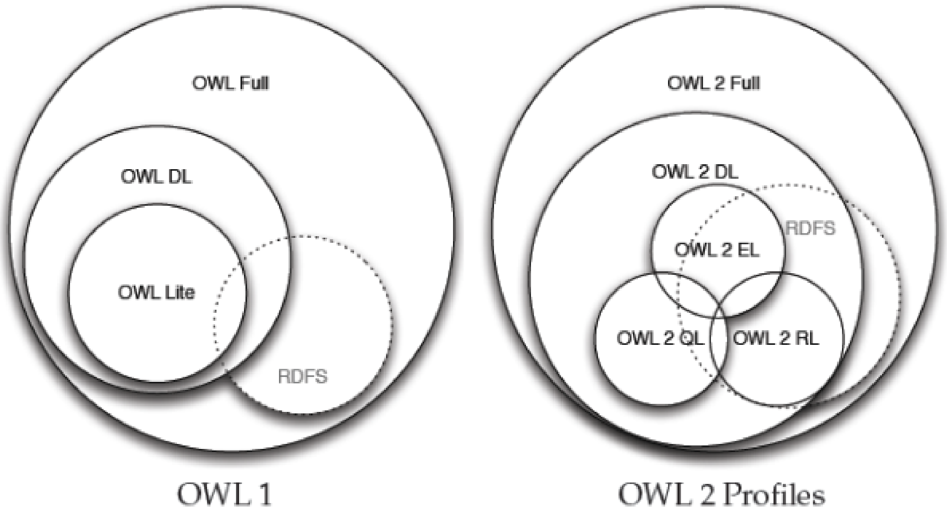


Figure 2.4: Graphical representation of the relation between RDF and OWL profiles

After its appearance, OWL has received several updates. In December 2006, OWL 1.1[10] added features requested by users: additional property and qualified cardinality constructors, extended datatype support, simple metamodelling, and extended annotations. This update moved OWL from the $\mathcal{SHOIN}$ DL that underlies OWL DL to the $\mathcal{SROIQ}$ DL. In December 2012, OWL 2[11] added new functionality including: keys; property chains; richer datatypes, data ranges; qualified cardinality restrictions; asymmetric, reflexive, and disjoint properties; and enhanced annotation capabilities [Grau et al., 2008]. OWL 2 also defines 3 new profiles where some of the restrictions applicable to OWL DL have been relaxed. Figure 2.4 shows the relations between OWL profiles and RDF.

- OWL 2 $\mathcal{EL}$ is particularly useful in applications employing ontologies that contain very large numbers of properties and/or classes.

[10] http://www.w3.org/Submission/owl11-overview/
[11] http://www.w3.org/TR/owl2-overview/

- OWL 2 $\mathcal{QL}$ is aimed at applications that use very large volumes of instance data, and where query answering is the most important reasoning task.
- OWL 2 $\mathcal{RL}$ is aimed at applications that require scalable reasoning without sacrificing too much expressive power.

OWL constructors and axioms

Tables 2.2 and 2.3 show some constructors available in OWL. Remember that axioms are the logical foundations of OWL. Every ontology component is represented in OWL with a constructor or a combination of them. OWL profiles differ, therefore, in the possible constructors to use.

Basic constructors		
`owl:Class`	`owl:Datatype`	`rdfs:range`
`owl:DatatypeProperty`	`rdfs:domain`	`rdfs:subClassOf`
`owl:imports`	`owl:ObjectProperty`	`owl:versionInfo`
`owl:Ontology`	`rdf:Property`	`rdfs:subPropertyOf`

Table 2.2: Basic OWL constructors

More complex constructors		
`owl:allValuesFrom`	`owl:maxCardinality`	owl:cardinality
`owl:complementOf`	`owl:maxCardinality`	`owl:differentFrom`
`owl:disjointWith`	`owl:onProperty`	`owl:FunctionalProperty`
`owl:hasValue`	`owl:someValuesFrom`	`owl:intersectionOf`
`owl:InverseFunctionalProperty`	`owl:TransitiveProperty`	`owl:inverseOf`
`owl:minCardinality`	`owl:Restriction`	`owl:SymmetricProperty`
`owl:unionOf`	`owl:oneOf`	`owl:equivalentClass`

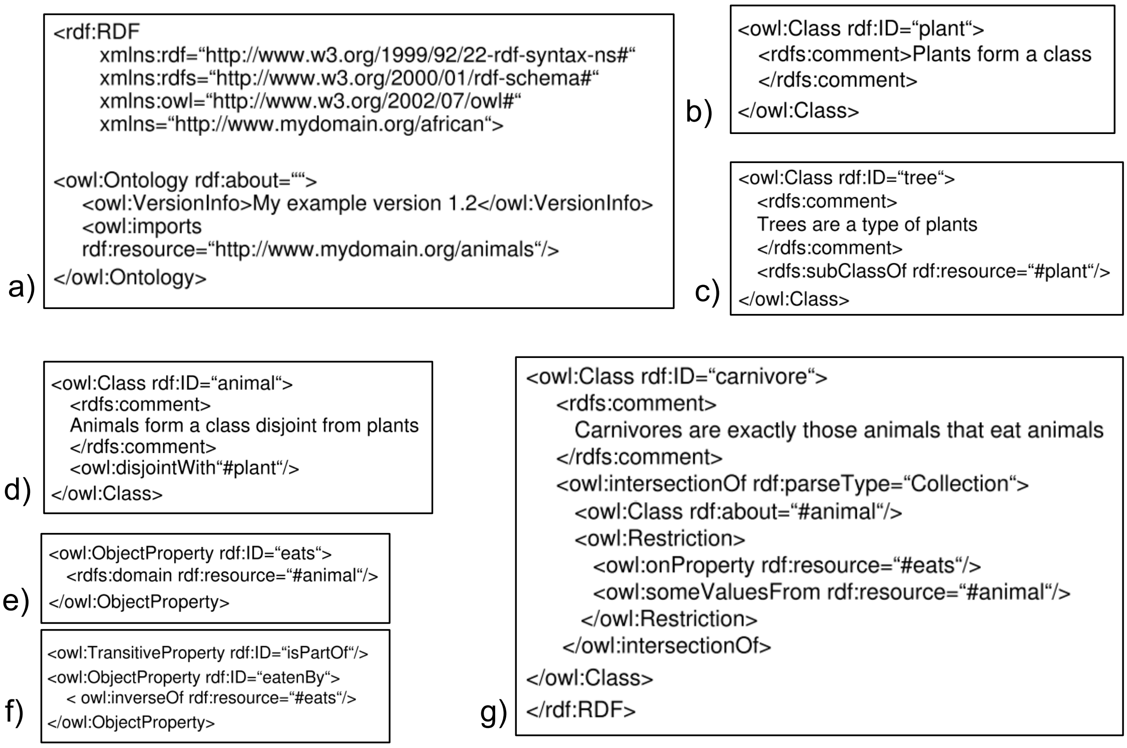
Table 2.3: Complex OWL constructors

Example: Using RDFS users can declare classes like `Country`, `Person`, `Student` and `Canadian`... Additionally, using OWL `Country` and `Person` can be defined as disjoint classes.

Example: Using RDFS users can state that `Canada` and `England` are both instances of the class `Country`... Additionally, using OWL `Canada` and `England` can be defined as different individuals.

Example of OWL components in OWL

Figure 2.5 shows some examples extracted from[12] about the use of OWL constructors for defining ontology components.



a)
```
<rdf:RDF
      xmlns:rdf=“http://www.w3.org/1999/92/22-rdf-syntax-ns#“
      xmlns:rdfs=“http://www.w3.org/2000/01/rdf-schema#“
      xmlns:owl=“http://www.w3.org/2002/07/owl#“
      xmlns=“http://www.mydomain.org/african“>

<owl:Ontology rdf:about=““>
   <owl:VersionInfo>My example version 1.2</owl:VersionInfo>
   <owl:imports
   rdf:resource=“http://www.mydomain.org/animals“/>
</owl:Ontology>
```

b)
```
<owl:Class rdf:ID=“plant“>
   <rdfs:comment>Plants form a class
   </rdfs:comment>
</owl:Class>
```

c)
```
<owl:Class rdf:ID=“tree“>
   <rdfs:comment>
   Trees are a type of plants
   </rdfs:comment>
   <rdfs:subClassOf rdf:resource=“#plant“/>
</owl:Class>
```

d)
```
<owl:Class rdf:ID=“animal“>
   <rdfs:comment>
   Animals form a class disjoint from plants
   </rdfs:comment>
   <owl:disjointWith“#plant“/>
</owl:Class>
```

e)
```
<owl:ObjectProperty rdf:ID=“eats“>
   <rdfs:domain rdf:resource=“#animal“/>
</owl:ObjectProperty>
```

f)
```
<owl:TransitiveProperty rdf:ID=“isPartOf“/>
<owl:ObjectProperty rdf:ID=“eatenBy“>
   < owl:inverseOf rdf:resource=“#eats“/>
</owl:ObjectProperty>
```

g)
```
<owl:Class rdf:ID=“carnivore“>
   <rdfs:comment>
     Carnivores are exactly those animals that eat animals
   </rdfs:comment>
   <owl:intersectionOf rdf:parseType=“Collection“>
     <owl:Class rdf:about=“#animal“/>
     <owl:Restriction>
       <owl:onProperty rdf:resource=“#eats“/>
       <owl:someValuesFrom rdf:resource=“#animal“/>
      </owl:Restriction>
   </owl:intersectionOf>
</owl:Class>
</rdf:RDF>
```

Figure 2.5: Example of OWL ontology components using OWL constructors and axioms

- Figure 2.5 a) defines an ontology that reuses another ontology through the constructor `owl:imports`; remember that one of the principles of ontologies is to be reusable.

- Figure 2.5 b) defines a concept `plant` and add a comment to this class using the constructor `rdfs:comment`.

- Figure 2.5 c) uses the constructor `rdfs:subClassOf` for defining a taxonomical relation (isA) between `tree` and `plant`.

[12] http://wwwdh.cs.fau.de/IMMD8/Services/textfarm/referate/RDF_und_OWL.pdf

- Figure 2.5 d) makes use of the constructor `owl:disjointWith` between the concepts `animal` and `plant`.

- Figure 2.5 e) and f) indicate that the relations `eats` and `eatenby` have transitive properties.

- Figure 2.5 g) defines the class `carnivore` that conceptually represents *those animals that eat animals* in this domain.

It should be noted that both Figure 2.1 and 2.5 are ontologies as both fit to Grubers's definition, however, just Figure 2.5 fit to Borst's and Guarino's definition. For further details about OWL, we recommend reading the chapter 8 in [Singh and Huhns, 2004].

2.1.5. Reasoning and inferencing

Formal semantics and the availability of efficient and provable correct reasoning tools have made the OWL DL dialect of OWL the language of choice for ontology development in fields as diverse as biology, medicine, geography, inter alia [Golbreich et al., 2007]. Formal semantics allows ontologies and information using vocabulary defined by ontologies, to be shared and exchanged without disputes as to precise meaning. The standardisation of OWL has sparked the development and/or adaption of a number of reasoners, including FacT++ [Tsarkov and Horrocks, 2006], Pellet [Sirin et al., 2007], HermiT [Shearer et al., 2008] or many others[13].

The more expressive a language is, the less decidable it is. Therefore, it depends on the needs of the system using ontologies to decide which KRL and reasonser to use for codifying and reasoning their ontologies. For example, due to the size of ontologies like SNOMED CT less expressive profiles like the $\mathcal{EL}$ are gaining popularity because more expressive profiles make the reasoning process computationally difficult and take more time than is desirable. The performance of a reasoner over one ontology will depend both the size and expressivity of the ontology and reasoner [Dentler et al., 2011].

[13] `http://owl.cs.manchester.ac.uk/tools/list-of-reasoners/`

2.2. Biomedical knowledge resources

Biomedical knowledge resources encompass many different types of data that are used in two disciplines: bioinformatics and clinical informatics.

2.2.1. Bioinformatics

The origin of bioinformatics can be set at the beginning of the 1960s [Hagen, 2000] as a consequence of the convergence of DNA sequencing, large-scale genome projects, the internet and supercomputers. In [Luscombe et al., 2001], bioinformatics is defined as:

> *"Bioinformatics is conceptualizing biology in terms of macromolecules (in the sense of physical-chemistry) and then applying "informatics" techniques (derived from disciplines such as applied maths, computer science, and statistics) to understand and organise the information associated with these molecules, on a large-scale."*

Between 1945 and 1955 Frederic Sanger and his team achieved the sequencing of a whole protein of insulin, which was codified as a sequence of amino-acids that define the structure of this protein in the DNA [Sanger, 1959]. Genes are transcribed into segments of RNA (ribonucleic acid), which are translated into proteins. Both RNA and proteins are products of the expression of the gene[14]. Then to sequence a protein is the first step to link this with: the fragment of DNA that codifies it, its function, other proteins/genes related, and many other pieces of information with a biological interest. Sanger's discovery triggered the appearance of a collection os amino-acids sequences that were used as sources of data in new research. This collection has grown and it is still growing exponentially; so its control went soon far from manual human techniques[15]. In general, bioinformatics has a three-fold aim [Luscombe et al., 2001]:

1. Organising data in a way that allows researchers to access existing information and to submit new entries as they are produced.

2. Developing tools and resources that support data analysis.

[14] `http://www.ncbi.nlm.nih.gov/books/NBK5191/def-item/gene-product/`

[15] `http://www.ncbi.nlm.nih.gov/genbank/statistics`

3. Using these tools to analyze the data and interpret the results in a biologically meaningful manner.

Characteristics of biological information

According to [Stevens and Lord, 2009] biological data are characterised in the following ways: (1) large quantity, (2) complexity, (3) volatility, (4) heterogeneity and (5) distribution. This scene leaves both the curators of bioinformatics resources and their users with great difficulties. A typical user, as well as a bioinformatics tool builder, is left trying to deal with the following problems in order to attempt tasks like: knowing which resources to use in a certain task, understanding the content of the resources and interpreting results, codifying in a computer the results of a physical experiment, and so on.

According to [Legaz-García, 2015], as a result of the codification of the results in a computer, several reference databases with biological knowledge have arisen: nucleotide databases (i.e. Gene Bank [Benson et al., 2007], RefSeq [Pruitt, 2004]), protein databases (i.e. UniProt [Consortium, 2012]), protein structure databases (i.e. PDB [Berman et al., 2000]), genomes and maps databases (i.e. Ensembl [Hubbard et al., 2002]) and databases focus on a concrete organism (i.e. Mouse Genome Database [Bult et al., 2007]). The number of records in such as databases is exponentially growing due to: (1) experimental data is recorded, and (2) bioinformatics techniques make use of computer for processing the available biological information. As a result, computer programs infer new knowledge that, in turn, is recorded in such a database. The problem is, how to sort such an amount of data with no consensus between humans and computers?

Gene Ontology and the success of ontologies in bioinformatics

According to their definitions: (1) *"...bioinformatics is conceptualising biology in terms of macromolecules..."*, and (2) ontologies are *"... specification of a conceptualisation..."*. So, in the biology domain ontologies provide a reference, structured and controlled vocabularies. The most successful ontology is the Gene Ontology taking into account both the number of users and the reach across species and granularities [Ashburner et al., 2000, Smith et al., 2007]. The GO project started in 1998. The GO describes gene products in terms of their associated biological processes, cellular components and molecular functions in a species-independent manner. As it was stated in [Consortium, 2001], the strength of GO approach lies in:

- It compiles a comprehensive structured vocabulary of terms describing different elements of molecular biology that are shared among life forms.
- Its focus on the specifics of the biological vocabularies and on the establishment of precise, defined logical relationships between the concepts (using a formal KRL).
- Its structure permits the implementation of robust query capabilities far beyond the development of a simple dictionary of terms or keywords (using reasoners).

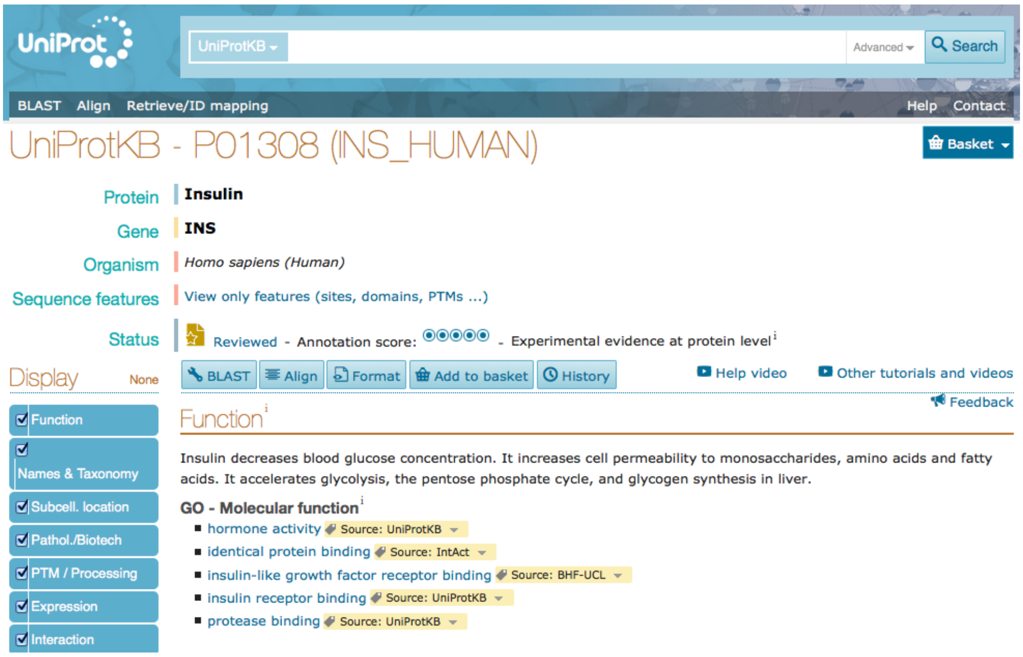
As it can be seen, these strengths are closely related to the definition of the GO as a formal ontology. For example, using meronymy relations the GO models `'DNA replication'` as part of `'DNA metabolism'` and as a part of `'DNA replication and Cell cycle'`, which is itself a part of the `'Cell cycle'`. Then researchers might describe new gene products, which are stored in databases, using references to GO concepts through unique IRIs (Internationalised Resource Identifiers). This process is known as the annotation process.

For example, Figure 2.6 shows a UniProt[16] record that contains information about the insulin human protein. UniProt is a database that contains comprehensive, high quality and freely accessible protein sequence and functional information. In the bottom part of this figure we can see annotations that use concepts defined in the GO. The annotation process can be both automatic and manual, so we see that each protein has an annotation score associate. Nowadays there are more than 900 000 annotations using GO terms in UniProt [Consortium, 2015]. Furthermore, more than 5 million GO annotations are distributed in other 32 databases[17].

Thus, the assignment of a uniquely defined GO concepts as an attribute of gene products, which is performed with the annotation process, also allows a subsequent query, via the defined concept, to recover all gene products known to share that attribute. For example, it is possible to query all UniProt entries annotated with the molecular function represented by the concept `'protease binding'`. Having the GO codified using a

[16] `http://www.ebi.ac.uk/uniprot`

[17] `http://geneontology.org/page/current-go-statistics`



UniProtKB - P01308 (INS_HUMAN)

Protein | Insulin
Gene | INS
Organism | *Homo sapiens (Human)*
Sequence features | View only features (sites, domains, PTMs ...)
Status | Reviewed - Annotation score: ●●●●● - Experimental evidence at protein level

Function

Insulin decreases blood glucose concentration. It increases cell permeability to monosaccharides, amino acids and fatty acids. It accelerates glycolysis, the pentose phosphate cycle, and glycogen synthesis in liver.

GO - Molecular function

- hormone activity Source: UniProtKB
- identical protein binding Source: IntAct
- insulin-like growth factor receptor binding Source: BHF-UCL
- insulin receptor binding Source: UniProtKB
- protease binding Source: UniProtKB

Figure 2.6: Entry in UniProt of the protein “insulin” for human (`http://www.uniprot.org/uniprot/P01308`)

formal KRL enables the development of queries based on such a formalism is possible. These query systems make use of reasoners that exploit the semantic within GO for obtaining the results. Continuing with the example, a query about entries annotated with the concept `binding` would retrieve all those entries annotated with it or any of its descendants like `‘protease binding’`. Moreover, the enrichment of `‘protease binding’` with the axiom `‘protease’ enables some ( ‘binds’ some ‘protease binding’ )` would query systems to retrieve all the entries annotated with bindings that are enabled by some `‘protease’`. It should be pointed out that this thesis wishes to contribute to creating richer ontologies, but we have not studied the impact of richer ontologies on annotations and query systems. However, the previous examples motivate with a practical example the benefits of more axiomatic and richer ontologies.

The success of the GO provoked the appearance of other biological ontologies focusing on different subdomains of biology, so ontologies were supported by biologies and other researches in the community. In particular, the Open Biomedical Ontologies (OBO) Foundry [Smith et al., 2007] is a coordinated initiative, which has contributed to building an orthogonal set of "good" ontologies that cover different aspects of biology, as well as to supporting biomedical data integration. The Sequence Ontology [Eilbeck et al., 2005], Cell Ontology or the ChEBI ontology are just a few examples.

2.2.2. Clinical informatics and ontologies

Clinical Informatics is defined as the application of informatics and information technology to deliver healthcare services. It is also referred to as applied clinical informatics and operational informatics[18]. Semantic Web Technologies have been considered fundamental for the achievement of semantic interoperability of clinical data as it was stated in the report "Semantic Interoperability for Better Health and Safer Healthcare" [Stroetman et al., 2009] or it is proposed by the Semantic HealthNet[19]. For example, SNOMED CT is used for annotating Electronic Health Records (EHR) so that Semantic Web Technologies support the communication between different systems across the world [Martínez-Costa, 2011].

2.2.3. Biomedical knowledge resources

As we have commented, the development and use of ontologies is now a mainstream activity within biology and medicine. Using ontology search engines like Watson [D'Aquin and Motta, 2011] or Swoogle [Ding et al., 2004] we can access to thousands of ontologies from any domain. More recently, the use of search engines is being replaced by repositories that stores ontologies like TONES Ontology Repository[20] or Ontohub [Mossakowski et al., 2014]. In contrast to others, TONES is a curated repository, so it is designed to be a central location for ontologies that might be of use to tools developers for testing purposes.

> *Digital curation*[21] *is the selection, preservation, maintenance, collection and archiving of digital assets. Digital curation establishes, maintains and adds value to repositories of digital data for present and future use. This is often accomplished by archivists, librarians, scientists, historians, and scholars.*

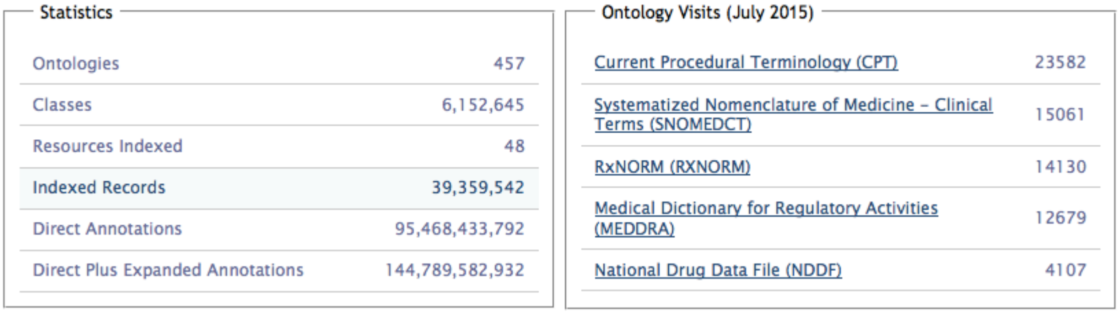
[18] https://www.amia.org/applications-informatics/clinical-informatics

[19] http://www.semantichealthnet.eu/

[20] http://owl.cs.manchester.ac.uk/repository/

[21] https://en.wikipedia.org/wiki/Digital_curation

In particular, we focus our attention in the analysis of biomedical ontology repositories like BioPortal [Whetzel et al., 2011a], which is managed by the National Center for Biomedical Ontology (NCBO) with the goal of expediting communication between researchers and biomedical ontology developers [Musen et al., 2012]. In the time of this writing it contained more than 450 ontologies (see Figure 2.7).



Statistics

Ontologies	457
Classes	6,152,645
Resources Indexed	48
Indexed Records	39,359,542
Direct Annotations	95,468,433,792
Direct Plus Expanded Annotations	144,789,582,932

Ontology Visits (July 2015)

Current Procedural Terminology (CPT)	23582
Systematized Nomenclature of Medicine - Clinical Terms (SNOMEDCT)	15061
RxNORM (RXNORM)	14130
Medical Dictionary for Regulatory Activities (MEDDRA)	12679
National Drug Data File (NDDF)	4107

Figure 2.7: Statistics describing BioPortal ontologies

Although BioPortal is probably one of the most popular biomedical ontology repositories, others exist like Aber-OWL [Hoehndorf et al., 2015, Slater et al., 2015]. Aber-OWL includes BioPortal ontologies, but the benefit of using Aber-OWL is that it provides access to ontologies that have been processed by reasoners. This avoids users having to dealing with technical problems associated with reasoning processes. Therefore Aber-OWL takes advantage of the expressivity of each ontology for classifying them or making other kind of queries over the inferred model.

It should be noted that these repositories could contain ontologies that are controlled vocabularies or plain taxonomies created by domain experts with low axiomatisation. Moreover, BioPortal and other biomedical repositories are not curated, so the applications of quality assurance methods could contribute to increment its quality. For example, in [Kamdar et al., 2015] reuse and overlapping in BioPortal ontologies is studied.

KRLs and biomedical ontologies

For biomedical ontologies, formal semantics and the availability of efficient and probably correct reasoning tools have made the OWL DL dialect of OWL the language of choice for ontology development. As a consequence, OWL DL is extensively used in the life sciences community, where it has rapidly become a *de facto* standard for ontology development and data interchange. For example, see BioPAX,2 NASA's SWEET ontologies and the National Cancer Institute Thesaurus [Grau et al., 2008].

However, OWL DL is not unique. The OBO Foundry community developed in parallel with the first version of OWL the OBO Flat File Format[22], which was revised in 2004 and received a later revision in 2006. In its origins the OBO format made possible to make a property reflexive and/or (anti-)symmetric and also let to make one property "transitive over" another. These axioms were not available in the first version of OWL. For this reason the OBO specification pointed out that OBO is a subset of the concepts in the OWL 1 DL, with several extensions for meta-data modelling and the modelling of concepts. However, OWL 1.1 is fully backwards compatible with OWL so methods for mapping between OBO and OWL and vice-versa have been in development [Golbreich et al., 2007].

OWL and OBO are the most used KRLs in BioPortal being the distribution in September 2015: OBO (103), OWL (295), SKOS (1) and UMLS (33). As just 7.8 % of biomedical ontologies use SKOS [Miles et al., 2005] and UMLS [Bodenreider, 2004] we exclude them from this study. Therefore, we consider as *biomedical knowledge resources* to biomedical ontology repositories, and in particular those ontologies that use OBO or OWL as KRL.

2.3. Ontologies and identifiers in natural language

Concepts are classes in OWL that use the `owl:Class` constructor for their definition (Figure 2.5 b). Taxonomical relations are set using `owl:subClassOf` and `owl:EquivalentClass` constructors (Figure 2.5 c). The desirable situation is that classes in OWL are defined by stating properties of concepts and relations between them. In OWL associative relations between classes (also known as roles or properties) are defined using the `owl:ObjectProperty` constructor (Figure 2.5 g).

[22] `http://oboformat.googlecode.com/svn/trunk/doc/GO.format.obo-1_2.html`

Identifiers and natural language

So far, we have described ontologies as well as the importance of codifying them using a formalism that makes them machine friendly. However, humans need to be able to understand the conceptualisation when they create, read or expand an ontology. Humans use the natural language for communications so pieces or fragment of it are included in ontologies for making them human friendly.

In OWL, a named entity refers to a named class, a named individual or a named property. Each named entity must have a unique identifier, called an IRI (Internationalized Resource Identifiers). Ontologies have a base IRI that identifies it across the internet. For example, the Gene Ontology has an IRI base that unequivocally references to it. Moreover, each class, relation or instance in such ontology would have one IRI that share the base IRI of the main ontology. In the next line we highlight in blue the IRI base of the Gene Ontology, and in brown the identifier of one of its classes.

http://purl.obolibrary.org/obo/GO_0005488

An IRI can also address a particular element within an XML document by including an IRI fragment identifier as part of the IRI. An IRI which includes an IRI fragment identifier consists of an optional base IRI, followed by a "#" character, followed by the IRI fragment identifier.

http://purl.org/obo/owl/GO#GO_0005488

In these two examples, they use GO_0005488 as identifier. This identifier is meaningless, which is also called a "semantic-free" identifier. This identifier is meaningless because the identifier has no direct relationship between the textual description and the characteristics about the entity being identified.

Although IRIs contribute to the semantic interoperability of data that use ontologies for modelling a domain, natural language descriptions help domain experts in a better understanding of the ontology content. For these reason ontology builders use fragments in natural language as fragments of the IRIs of ontology components.

http://www.co-ode.org/ontologies/galen#Binding

However, the use of natural language in IRI fragments requires them to fit to the restrictions of the syntax. In [Nor Azlinayati Abdul et al., 2010] is presented a survey of the usage and style of identifiers and labels of named entities in a corpus of OWL ontologies. They identify 7 lexical encoding styles: (1) CamelCaseStyle, (2) Underscore_style, (3) Hyphenstyle, (4) HybridCamelCase_underscore_style, (5) HybridCamelCasehyphenstyle, (6) Hybridhyphen_underscore_style, and (7) single word. According to this survey, the CamelCaseStyle is the most widely used for identifiers.

Annotations in OWL

In OWL it is possible to separate the IRI for the entity and the label for that entity. OWL ontologies and entities can be assigned annotations, which are pieces of extra-logical information describing the ontology or entity using natural language. For example, Figure 2.5 shows some comments in natural language for the different classes: *"plants form a class"*, *"trees are a type of plants"* or *"carnivores are exactly those animals that eat animals"*. Annotations in OWL are written using annotation properties: `owl:versionInfo`, `rdfs:label`, `rdfs:comment`, `rdfs:seeAlso`, and `rdfs:isDefinedBy`. Ontology engineers can also create their own annotation properties; for example, SNOMED CT allows three types of labels: *fully specified name* (FSN), *preferred term* (PT) and *synonym* (S). It should be noted that annotations are extra-logical constructs because adding or removing them should not affect the set of consequences derivable from an ontology [Grau et al., 2008].

rdf:comment vs rdf:label

However, although the range of annotations properties is strings, not all of them must contain natural language descriptions. We focus our attention on two of them: `rdf:comment` and `rdf:label`. They usually contain descriptions in natural language, although they have different purposes. On the one hand, comments include whatever information about the entity for which they are associated. On the other hand, labels should describe without ambiguity the classes that they represent; labels are usually nominal phrases that let users understand the contextualisation that the ontology object represents. Labels are also known as the names for predicates and constants in rules or logical formulas, and they constitute an ontological vocabulary [Stephan et al., 2007].

Therefore, labels can be considered as textual identifiers that complement the logical identifier codified in the URI.

How do ontologies define identifiers?

The survey made in [Nor Azlinayati Abdul et al., 2010] concluded that most ontologies do not use labels for named entities, but when they do use labels, these labels are mostly meaningful.

2.3.1. Guidelines for naming identifiers

Common naming conventions also facilitate understanding the meaning of the content of the ontology. In particular, inconsistencies in naming conventions can impair the readability and navigability of ontology class hierarchies, and hinder their alignment and integration [Schober et al., 2009]. In order to achieve a common naming convention, the OBO Consortium promotes principles as models of good practice[23].

Among other principles, they propose the use of naming conventions. They group 16 naming conventions in four global groups: (1) be clear and unambiguous, (2) be univocous, (3) reduce string variance and (4) align typography. Some of these principles were obtained in [Schober et al., 2009] using a survey carried out to establish which naming conventions were employed by OBO Foundry ontologies. The application of unified naming conventions will help to harmonise the appearance and increase the robustness of concepts within ontologies.

Naming conventions also ease the application of computerised lexical analysis and processing[24]. For example, the use of systematic naming so that a subclass contains in its definition part of the father should indicate a `rdfs:subClassOf` relation. For example, the `rdfs:subClassOf` relation between the morphologic abnormalities `‘Congenital stenosis’` and `‘Stenosis’` is lexically present too. Both classes follow a systematic naming so the more specific class contains in its natural language description part of the parent.

[23] `http://www.obofoundry.org/crit.shtml`

[24] `http://wiki.obofoundry.org/wiki/index.php/Naming`

2.3.2. Hidden semantics

Machines cannot easily exploit the knowledge expressed only as text, which limits the usefulness of such ontologies. In [Third, 2012] a distinction between two types of identifiers is made:

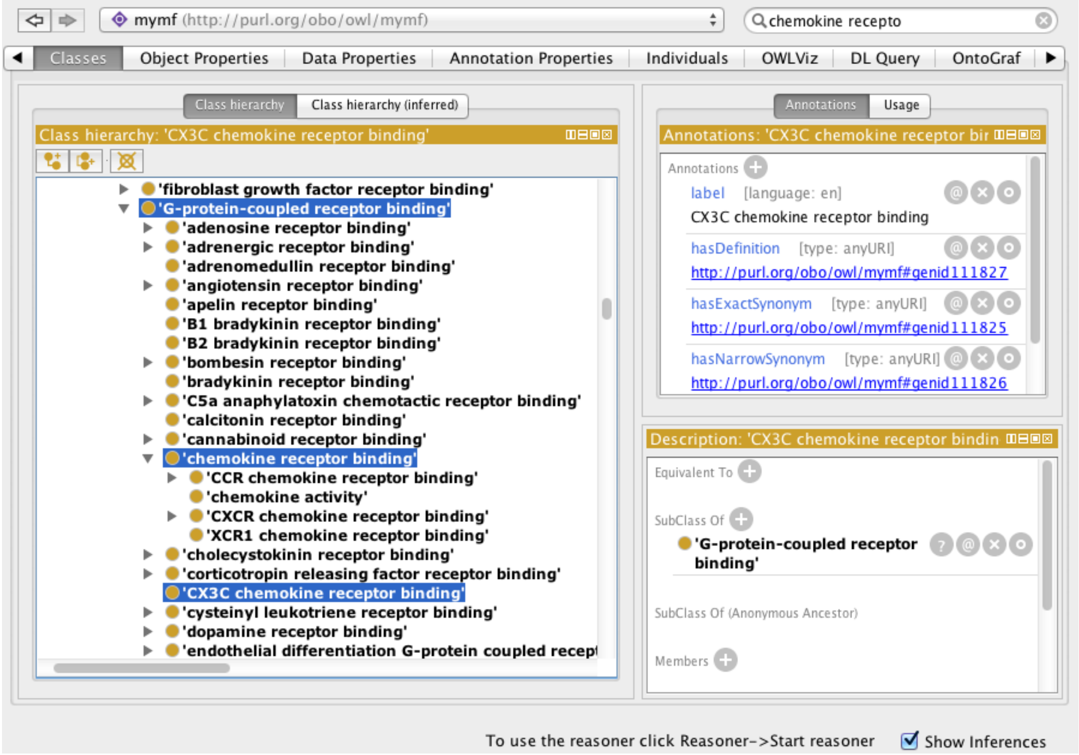
- *Simple identifier:* this is an identifier that consists of a single natural language word. For example, `Congenital` or `Stenosis`.
- *Complex identifier:* this is an identifier that consists of multiple natural language words. For example, `'Congenital Stenosis'`.

After this, a *constructed identifier* is defined as a *complex identifier* where its component words (or just its content words) are themselves simple identifiers in the containing ontology. For example, if an ontology contains identifiers corresponding to `'Congenital'`, `'Stenosis'` and `'Congenital Stenosis'` then `'Congenital Stenosis'` is a constructed identifier. Then, the meaning of a constructed identifier can be *defined* in an ontology by axioms in which all, or most of its component or content words occurs as, or in, identifiers. Third uses these ideas to extract a list of the 10 most frequent patterns for defining axioms using a corpus of 548 OWL ontologies (see Table 2.4).

From the figures of Table 2.4 it can be concluded that ontology developers follow a systematic naming when they name ontology components. This is best practice that according to [Power, 2010] the vast majority do. However, biomedical ontologies like GO were considered to have *hidden semantics*, which means that some knowledge is expressed as identifiers but not as axioms [Wroe et al., 2003, Egaña Aranguren et al., 2008, Fernandez-Breis et al., 2010, Mungall et al., 2011]. In the context of Third´s patterns, *hidden semantics* might be detected if we find ontology components that follow the lexical pattern in the identifier (second column in Table 2.4) but do not follow the axiomatic pattern (third column in Table 2.4). Figure 2.8 shows an example of this *hidden semantics*, which has been found in the version of the Gene Ontology Molecular Function enriched in [Fernandez-Breis et al., 2010]. The class `'3X3C chemokine receptor binding'` is a *complex identifier* as it can be decomposed in multiple words. Moreover, `'chemokine receptor binding'` is a *complex identifier* too. Then they lexically follow the pattern *DCBA CBA* in the identifier but the axiomatic pattern `subClassOf(DCBA, CBA)` is not followed.

Freq.	Pattern	Example
1430	SubClassOf(AB B)	SubClassOf(representation-activity activity)
1179	SubClassOf(ABC BC)	SubClassOf(Quantified set builder Set builder)
455	InverseObjectProperties(hasA isAof)	InverseObjectProperties(HasInput IsInputOf)
387	SubClassOf(ABCD BCD)	SubClassOf(Continental-Statistical-Water-Area Statistical-Water-Area)
348	SubClassOf(ABCD CD)	SubClassOf(NonWikipediaWebPage WebPage)
240	SubClassOf(ABC AC)	SubClassOf(Process-Resource-Relation Process-Relation)
229	ObjectPropertyRange(hasA A)	ObjectPropertyRange(hasAnnotation Annotation)
192	ObjectPropertyRange(hasAB AB)	ObjectPropertyRange(hasTrustValue TrustValue)
188	InverseObjectProperties(AB ABof)	InverseObjectProperties(situation-place situation-place-of)
179	InverseObjectProperties(Aof hasA)	InverseObjectProperties(contentOf hasContent)

Table 2.4: 10 most frequent patterns of defining axioms


mymf (http://purl.org/obo/owl/mymf)
chemokine recepto
Classes
Object Properties
Data Properties
Annotation Properties
Individuals
OWLViz
DL Query
OntoGraf
Class hierarchy
Class hierarchy (inferred)
Class hierarchy: 'CX3C chemokine receptor binding'
'fibroblast growth factor receptor binding'
'G-protein-coupled receptor binding'
'adenosine receptor binding'
'adrenergic receptor binding'
'adrenomedullin receptor binding'
'angiotensin receptor binding'
'apelin receptor binding'
'B1 bradykinin receptor binding'
'B2 bradykinin receptor binding'
'bombesin receptor binding'
'bradykinin receptor binding'
'C5a anaphylatoxin chemotactic receptor binding'
'calcitonin receptor binding'
'cannabinoid receptor binding'
'chemokine receptor binding'
'CCR chemokine receptor binding'
'chemokine activity'
'CXCR chemokine receptor binding'
'XCR1 chemokine receptor binding'
'cholecystokinin receptor binding'
'corticotropin releasing factor receptor binding'
'CX3C chemokine receptor binding'
'cysteinyl leukotriene receptor binding'
'dopamine receptor binding'
'endothelial differentiation G-protein coupled recept
Annotations
Usage
Annotations: 'CX3C chemokine receptor bir
Annotations
Description: 'CX3C chemokine receptor bindin
Equivalent To
SubClass Of
SubClass Of (Anonymous Ancestor)
Members

label [language: en]
CX3C chemokine receptor binding

hasDefinition [type: anyURI]
http://purl.org/obo/owl/mymf#genid111827

hasExactSynonym [type: anyURI]
http://purl.org/obo/owl/mymf#genid111825

hasNarrowSynonym [type: anyURI]
http://purl.org/obo/owl/mymf#genid111826

SubClass Of: 'G-protein-coupled receptor binding'

To use the reasoner click Reasoner->Start reasoner ☑ Show Inferences

Figure 2.8: Hidden semantic in the class "CX3C chemokine receptor binding".

In [Mungall et al., 2011] is applied another method that elucidates *hidden semantics* using as a source two different ontologies. This lets us discover situations like: the class `'oocyte differentiation'`, from Gene Ontology, is a type of `'cell differentiation'` that is implicitly referencing in its label to the class `oocyte` in the Cell Ontology. This reasoning can be easily done by a human. However, if the class `‘ooccyte differentiation’` has no axioms that make explicitly this relation this knowledge will be out of the scope of the description logic reasoners.

2.3.3. Lexically-suggest logically define

In previous section we related, for the first time, lexical description and axiomatic description. Within the biomedical ontology literature, this idea is closely related to the idea introduced by Rector and colleagues: *“lexically suggest, logically define”* [Rector and Iannone, 2012]. They contributed to the quality assurance of SNOMED CT by analysing the axiomatic use of qualifiers like *chronic*, *acute*, *congenital* and so on. For example, given the classes `‘Congenital (qualifier value)’` and `‘Congenital stenosis (morphologic abnormality)’` there is a lexical suggestion, because they share the string *congenital*, so a logical relation between them should exist. Previously, in [Campbell et al., 1998] the *“lexically suggested logical closure”* metric was defined for medical terminology maturity. This metric was based on the evaluation of relationships that were proposed by lexical processing programs.

2.4. Ontology engineering

Ontology Engineering refers to the set of activities that concern the ontology development process, the ontology life cycle, as well as the methodologies, tools and languages required for building ontologies[25].

2.4.1. Methodologies for building ontologies

Since the appearance of ontologies a wide variety of methodologies for building ontologies as artifacts software has been proposed. As has been proposed in [Legaz-García, 2015], there are roughly 3 ways of building ontologies: (1) manual creation of ontologies from scratch, (2) collaborative and decentralised building of ontologies, and (3) reuse of ontologies.

[25] `http://mayor2.dia.fi.upm.es/oeg-upm/index.php/en/researchareas/2-ontologicalengineering`

Manual creation of ontologies

Many methodologies has been proposed for building ontologies from scratch [Lenat and Guha, 1989, Uschold and King, 1995, Gruninger and Fox, 1996, Fernandez-Lopez et al., 1999]. Although they differ in the number of steps, we can summarise the most important ones:

1. Identify the scope of the ontology [Uschold and King, 1995, Gruninger and Fox, 1996]. Make question in natural language for determining the scope, competency question [Gruninger and Fox, 1996].

2. Manual extraction of common implicit knowledge from several sources [Lenat and Guha, 1989].

3. Capture concepts, relations and term used for building the vocabulary [Uschold and King, 1995].

4. Codify the ontology [Uschold and King, 1995]. Specify the ontology in a formal language and define the competency question formally. Specify the axioms and definitions for the ontology terms formally. Set completeness conditions for the ontology [Gruninger and Fox, 1996].

5. Use of natural language tools or machine learning for acquiring new knowledge. First the result of these tools is processed automatically, and in a next step the process should be completely automatic [Lenat and Guha, 1989].

Moreover, Methontology [Fernandez-Lopez et al., 1999] proposes a life cycle based on the evolution of prototypes. This methodology permits the building of the ontology from scratch but also reusing existent ontologies directly or applying re-engineering. Methontology proposes 3 category levels (1) management activities, (2) development activities, (3) maintenance activities. Being the development activities those that are more closely related to the structure and knowledge formalism of the ontology.

Collaborative building of ontologies

Previous methodologies assume that the ontology is developed by a unique expert or team. However, the success of the biomedical ontologies has developed in the community [Malone and Stevens, 2013]. This collaborative effort makes it difficult to use rigid methodologies for building the ontology. In an ontology with hundreds or thousands of concepts and relations, even a group of domain experts could argue for reaching a shared contextualisation.

This flexibility is captured in the NeOn methodology [Suárez-Figueroa et al., 2012]. The NeOn Methodology does not prescribe a rigid workflow in contrast to other approaches. The NeOn methodology suggests a variety of pathways or scenarios for developing ontologies. Figure 2.9 shows these scenarios. The description of each scenario is taken from [Suárez-Figueroa et al., 2012]:

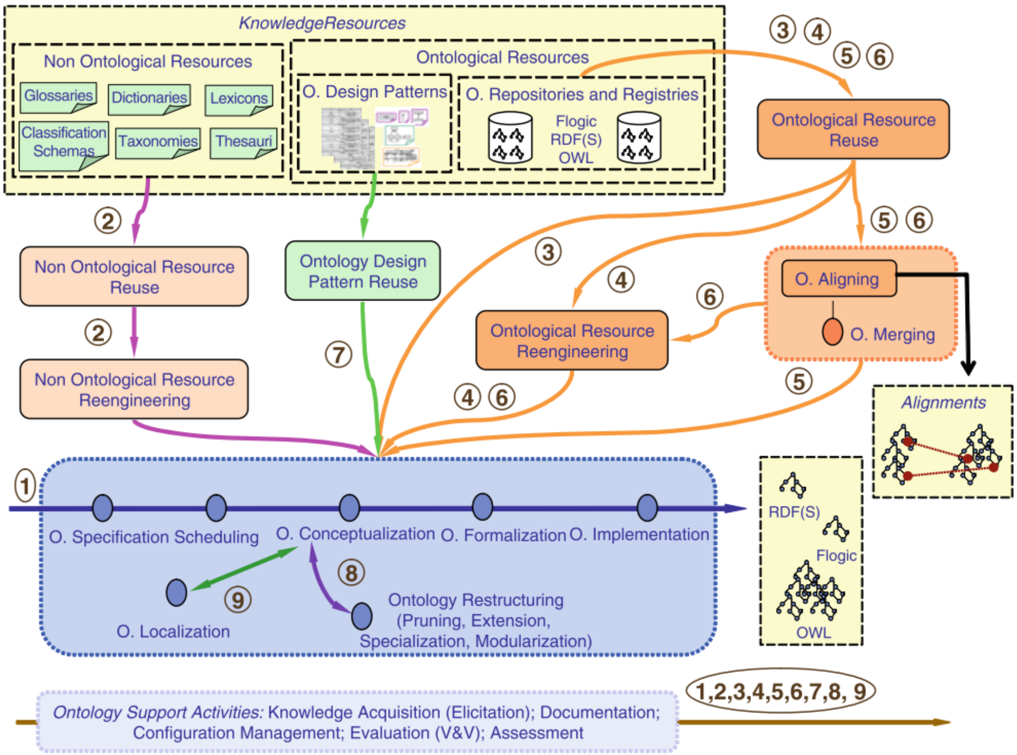

KnowledgeResources
Non Ontological Resources
Glossaries
Dictionaries
Lexicons
Classification Schemas
Taxonomies
Thesauri
Ontological Resources
O. Design Patterns
O. Repositories and Registries
Flogic
RDF(S)
OWL
3 4
5 6
Ontological Resource Reuse
2
Non Ontological Resource Reuse
Ontology Design Pattern Reuse
3
4
5 6
O. Aligning
O. Merging
6
Ontological Resource Reengineering
2
Non Ontological Resource Reengineering
7
4 6
5
Alignments
1
O. Specification Scheduling
O. Conceptualization
O. Formalization
O. Implementation
RDF(S)
Flogic
OWL
8
9
O. Localization
Ontology Restructuring (Pruning, Extension, Specialization, Modularization)
1,2,3,4,5,6,7,8, 9
Ontology Support Activities: Knowledge Acquisition (Elicitation); Documentation; Configuration Management; Evaluation (V&V); Assessment

Figure 2.9: Scenarios for building ontologies and ontology networks

- Scenario 1: from specification to implementation. The ontology network is developed from scratch, that is, without reusing available knowledge resources.

- Scenario 2: reusing and re-engineering non-ontological resources. This scenario covers the case where ontology developers need to analyze non-ontological resources and decide, according to the requirements the ontology should fulfil which non-ontological resources can be reused to build the ontology network. The scenario also covers the task of re-engineering the selected resources into ontologies.

- Scenario 3: reusing ontological resources. Here, ontology developers reuse ontological resources (ontologies as a whole, ontology modules, and/or ontology statements).
- Scenario 4: reusing and re-engineering ontological resources. Here, ontology developers both reuse and re-engineer ontological resources.
- Scenario 5: reusing and merging ontological resources. This scenario unfolds only in those cases where several ontological resources in the same domain are selected for reuse and when ontology developers wish to create a new ontological resource from two or more ontological resources.
- Scenario 6: reusing, merging, and re-engineering ontological resources. This scenario is similar to Scenario 5; however, here developers decide not to use the set of merged resources as it is, but to re-engineer it.
- Scenario 7: reusing ontology design patterns (ODPs) [Gangemi, 2005]. Ontology developers access ODPs repositories to reuse them.
- Scenario 8: restructuring ontological resources. Ontology developers restructure (i.e. modularising, pruning, extending, and/or specialising) ontological resources to be integrated in the ontology network being built.
- Scenario 9: localising ontological resources. Ontology developers adapt an ontology to other languages and culture communities, thus producing a multinligual ontology.

Although these scenarios can be combined in different and flexible ways, any combination of scenarios should include Scenario 1 because this scenario is made up of the core activities that have to be performed in any ontology development. The method developed in this thesis wants to contribute to scenarios 4, 5, 7 and 8, as it will be a way of assisting domain experts to evaluate concrete problems or pitfalls in their ontologies.

2.4.2. Ontology enrichment

Scenario 8 in the NeOn methodology proposes a situation where ontologies developers make changes in ontologies. One of the re-structurations is extending the ontology, which is related to *ontology enrichment* techniques, which are so close to *ontology learning* techniques.

- *Ontology enrichment* starts from a given ontology and has the aim of generating additional concepts or axioms using statistical data about the usage of the name of the concepts of the ontology in a text corpus [Brewster, 2006].

- *Ontology learning* is built upon well-established techniques from a variety of disciplines, including natural language processing, machine learning, knowledge acquisition and ontology engineering. Because the fully automatic acquisition of knowledge by machines remains in the distant future, the overall process is considered to be semi-automatic with human intervention [Cimiano et al., 2009].

The analysis of specialised text corpora makes it possible to automatically model certain domains through the construction of an ontology based on the content of the document in the corpora. This is achieved using Natural Language Processing (NLP) algorithms. Usually, ontology enrichment and ontology learning techniques also make use of NLP algorithms. The NLP requirements of ontology enrichment are also related to the interpretation of text [Navigli and Velardi, 2004, Friedman et al., 2006].

Natural Language Processing

NLP can be defined as *"a theoretically motivated range of computational techniques for analyzing and representing naturally occurring texts at one or more level of linguistic analysis for the purpose of achieving human-like language processing for a range of tasks or applications"* [Liddy, 2001]. According to [Ruiz-Martínez, 2011], people extract meaning from text of spoken language on at least seven levels: phonologica, morphological, syntactic, semantic, discourse and pragmatic. It should be pointed out that not all NLP systems use every level. Moreover, for each linguistic analysis level, NLP provides a set of tools in order to analyse the language [Rodríguez-García, 2014].

Ontology learning approaches [Liu et al., 2013] can be subdivided by extraction tasks: terms, synonyms, concepts, relations, and axioms. In particular, the acquisition, description and formalisation of semantical relations is an important requirement for increasing the expressivity of ontologies. While the automatic identification of terms from texts has been achieved [Uzuner et al., 2010], the extraction of semantical relations between the concepts is a bottleneck, as it is a large and tedious process that requires domain experts collaboration; being a process difficult to automatise. Among the different NLP techniques for discovering semantic relations from texts, we have focused on pattern-based text mining techniques. They were used for the first time by Hearst [Hearst, 1992] and have gained popularity step by step. These kind of techniques assume:

- The goal relation exists.
- The goal relation is a specific relation.
- The goal relation is explicitly expressed in the text.
- The goal relation can be detected by analyzing the words or lexical units.

For example, the patterns *"X like Y"* can be used to find fragments like *"Proteins like Insulin"*. Therefore, the process of extraction of semantical relations based on patterns implies the next steps:

1. Define the goal relation.
2. Discover real patterns that express the goal relation.
3. Find instances of the goal relation in the text using the patterns defined in step 2.
4. Use the patterns to create a new ontology or enrich other previously set.

Lexico-syntactic patterns [Hearst, 1992, Liu et al., 2011a] detect, and the exploitation of compound or multi-word terms may help to identify hierarchical relationships. Statistical approaches are based on Firth´s notion [Widdowson, 2007]; *"a word is characterised by the company it keeps"* so the analysis of co-occurrences of a word play an important role in the classification of such a word. [Liu et al., 2011b]. The CAMÉLÉON method [Aussenac-Gilles and Jacques, 2008] lets users to apply NLP techniques over a corpus of text and propose the user potential semantical relations taking as reference 71 patterns previous stored and organised as: (a) 19 patterns for definitions, (b) 35 patterns for hyperonymy, (c) 14 patterns for meronymy, (d) 1 pattern for reformulation, and (e) 2 *varia*. Then, CAMÉLÉON guides the user in the steps 2, 3 and 4 previously explained, so that the detection of patterns like *"X like Y"* is semi-automatised and the user can refined the initial patterns iterating over the method. Additionally, in the book *"Pattern-based Approaches to Semantic Relation Extraction"* [Auger and Barrière, 2008a], several methods are proposed. All of them follow a similar pattern-based text mining technique based on the mentioned 7 levels that compose a traditional NLP pipeline. The inclusion or exclusion of stages depends on the type of semantic relation to capture. The methods presented in [Auger and Barrière, 2008a] are applied both English and Spanish corpus, and they search both taxonomical and associative relations.

In the biomedical domain, one of the most active groups is BioNLP[26]. The goal of this community is to contribute to solving problems of the biomedical community that can be solved using NLP techniques. They focus their efforts on extracting relations from the study of biomedical scientific literature [Liu et al., 2012, Bravo et al., 2014]. For example, Figure 2.10 shows the discovery of an event (bind) and other related elements from the inspection of a corpus formed by biomedical texts.

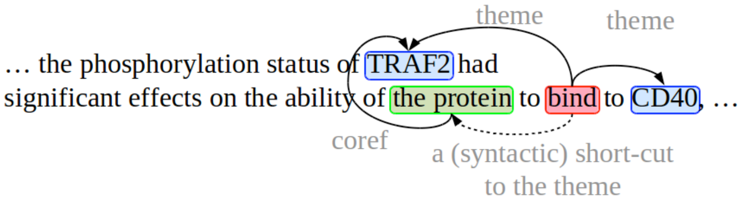

theme
theme
… the phosphorylation status of TRAF2 had
significant effects on the ability of the protein to bind to CD40, …
coref
a (syntactic) short-cut
to the theme

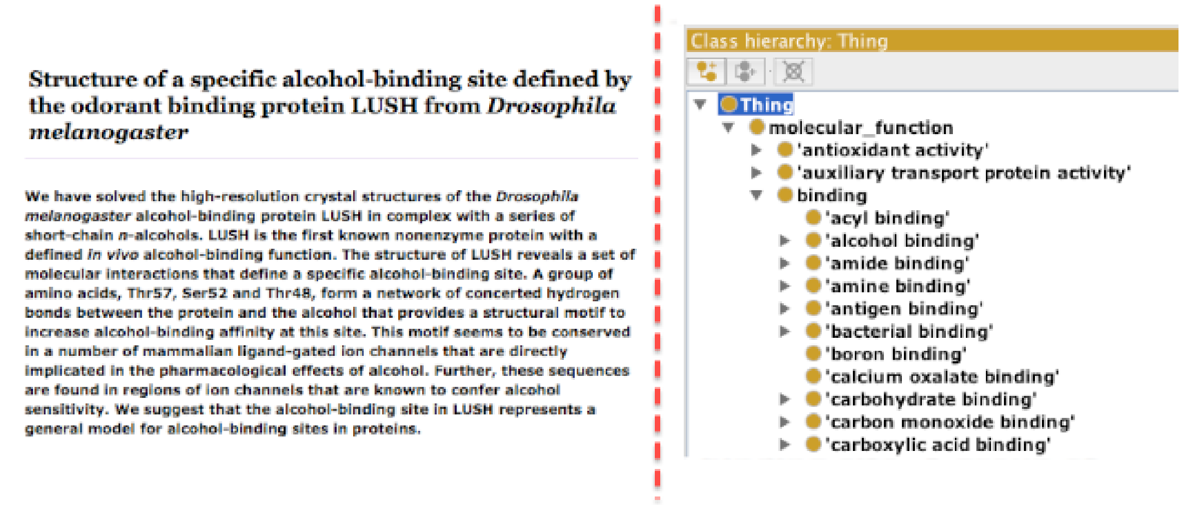
Figure 2.10: Example of annotations that search events and co-references in biomedical texts. Source `http://bionlp.dbcls.jp/redmine/projects/bionlp-st-ge-2013/wiki/Wiki`

In recent years, the utilisation of Machine Learning techniques [Nasraoui, 2008] in the relation extraction work seems to improve the results [Fauconnier et al., 2015]. In [Abney, 2007], a deep analysis of semi-supervised techniques applied in the computational linguistic scope is presensted.

[26] `http://www.bionlp.org/`

Ontology enrichment and biomedical ontologies

As we have commented, ontology enrichment and ontology learning techniques have been used for automating the development and maintenance of biomedical ontologies [Liu et al., 2011b]. However, they are mainly focused on analysing text that contributes to creating new ontological elements. For instance, the ODIE project[27] uses NLP techniques to identify and retrieve relevant free text information from clinical document repositories using ontological terminology, with the goal of improving and enriching ontologies. However, our scenario (ontology identifiers) is different as it is exemplified in Figure 2.11. Texts contain sentences with verbs, nouns, terms, pronouns and many other pieces of information (Figure 2.11 left), and ontology identifiers are usually nominal phrases that intend to provide an unambiguous way to name a concept (Figure 2.11 right). Given their different natures, the performance of traditional NLP methods for relation extraction is not the desired one.



Structure of a specific alcohol-binding site defined by the odorant binding protein LUSH from *Drosophila melanogaster*

We have solved the high-resolution crystal structures of the *Drosophila melanogaster* alcohol-binding protein LUSH in complex with a series of short-chain *n*-alcohols. LUSH is the first known nonenzyme protein with a defined *in vivo* alcohol-binding function. The structure of LUSH reveals a set of molecular interactions that define a specific alcohol-binding site. A group of amino acids, Thr57, Ser52 and Thr48, form a network of concerted hydrogen bonds between the protein and the alcohol that provides a structural motif to increase alcohol-binding affinity at this site. This motif seems to be conserved in a number of mammalian ligand-gated ion channels that are directly implicated in the pharmacological effects of alcohol. Further, these sequences are found in regions of ion channels that are known to confer alcohol sensitivity. We suggest that the alcohol-binding site in LUSH represents a general model for alcohol-binding sites in proteins.

Class hierarchy: Thing

- Thing
  - molecular_function
    - 'antioxidant activity'
    - 'auxiliary transport protein activity'
    - binding
      - 'acyl binding'
      - 'alcohol binding'
      - 'amide binding'
      - 'amine binding'
      - 'antigen binding'
      - 'bacterial binding'
      - 'boron binding'
      - 'calcium oxalate binding'
      - 'carbohydrate binding'
      - 'carbon monoxide binding'
      - 'carboxylic acid binding'

Figure 2.11: Different nature of an extract in natural language from a corpus of biomedical scientific documents and identifiers in GO MF ontology.

At this point, it should be pointed out that we do not use pattern-based text mining as we do not execute patterns against ontology labels in order to discover relations or use texts for discovering new content to include in the ontology. On the contrary, what

[27] http://bioontology.stanford.edu/ODIE-project

we do is to process ontology label with the goal of extract patterns that can be used to identify a relation. However, due to the popularity of pattern-based text mining for extracting relations, during this thesis we have tried them against a “artificial” document that contains each label in an ontology as single sentence. However, the performance was not good enough so we study in this field to methods that use information expressed in the ontology, which could be used to establish new formal relationships between existing ontologies, increasing the potential and usefulness of the biomedical applications that are supported by such ontologies [Golbreich et al., 2013]. In recent years, different approaches have been proposed within this research area:

- As we have mentioned, [Campbell et al., 1998] defined the *“lexically suggested logical closure”* metric for medical terminology maturity. This metric was based on the evaluation of relationships that were proposed by lexical processing programs.
- The Gene Ontology Next Generation project aimed to provide a method for the migration of biological ontologies to formal languages such as the Web Ontology Language (OWL) and to explore issues that are related to the maintenance of large biological ontologies [Wroe et al., 2003, Egaña Aranguren et al., 2008].
- The Open Bio-Ontology Language (OBOL) project [Mungall, 2004] generated formal relationships for existing OBO ontologies using reverse engineering. Later, [Bada and Hunter, 2007] described a frame-based integration of the GO and two other ontologies for improving the logical axioms between classes of biological concepts.
- Additionally, [Fernandez-Breis et al., 2010] proposed a method for the enrichment of ontologies by defining ontology design patterns [Gangemi and Presutti, 2009] and their corresponding implementation in the Ontology Pre-Processor Language[28].
- [Mungall et al., 2011] addressed the normalisation of GO by explicitly stating the labels of the compositional classes and partitioning them into mutually exclusive cross-product sets; they used a combination of OBOL [Mungall, 2004] and manual curation to generate logical axioms, which they called logical definitions, for selected parts of GO.

[28] `http://oppl2.sourceforge.net/`

- [Pacheco et al., 2009] detected *hidden semantics* that is named underspecification in classes from the SNOMED CT without logical axioms; the authors used natural language processing, which associated each class with a set of equivalence classes that grouped lexical variants (based on their labels), synonyms and translations.

- [Golbreich et al., 2013] represented the Foundational Model of Anatomy ontology [Rosse and Mejino, 2003] in OWL2, exploiting the naming conventions in its labels to make explicit some *hidden semantics*. For example, the pattern *"A of B"* was used to enrich the class `'Lobe of Lung'`. In most cases, the name *"A of B"* is a contraction that is formed from *"A and B"* that omits some logical axiom p that relates the two entities, A and B. The missing p was recovered from scanning the list of property restrictions that are attached to the class. For example, `regional_part_of` is the p for `'Lobe of Lung'`.

2.4.3. Ontology matching

Ontology alignment, or ontology matching, is the process of determining correspondences between concepts. A set of correspondences is also called an alignment. Ontology matching is a solution to the semantic heterogeneity problem. It finds correspondences between semantically related entities of ontologies. For example, Figure 2.12 left shows an alignment between two simple ontologies.

According to [Euzenat and Shvaiko, 2011], the matching operation determines an alignment A' for a pair of ontologies O_1 and O_2 (see Figure 2.12 right). Hence, given a pair of ontologies, which can be very simple and contain one entity each, the matching task is finding an alignment between these ontologies (see Figure 2.12 left). There are some other parameters that can extend the definition of matching, namely: (i) the use of an input alignment A, which is to be extended; (ii) the matching parameters, for instance, weights, or thresholds; and (iii) external resources, such as common knowledge and domain specific thesauri. [Shvaiko and Euzenat, 2013] discuss approaches that come from semantic web and artificial intelligence as well as from databases.

A Lexical Matcher creates equivalence mappings between classes that have identical labels or synonyms [Faria et al., 2013b, Faria et al., 2013a]. A Lexical Matcher is one of the simplest and most efficient matching algorithms. One type of Lexical Matcher is the full-name matching algorithm, which is usually a standard first step in ontology

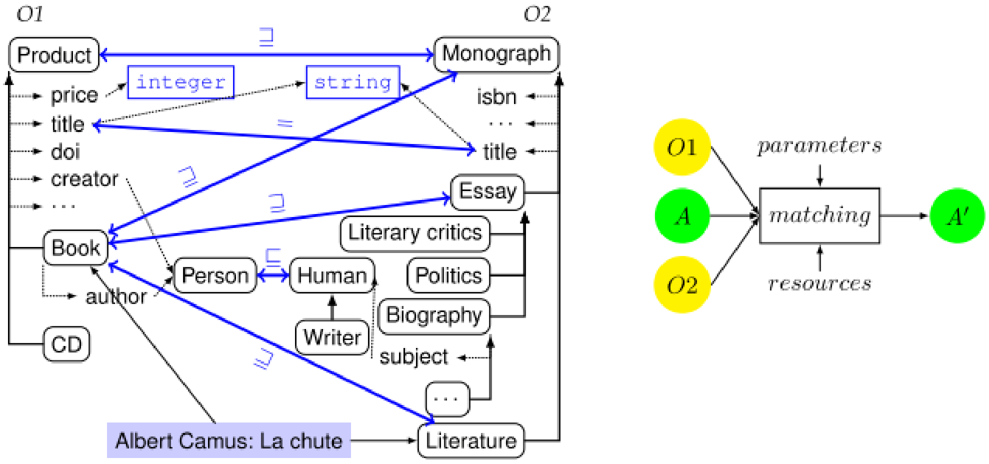


Figure 2.12: In the left part, two simple ontologies and an alignment. In the right part, the ontology matching operation. Figures taken from [Euzenat and Shvaiko, 2011].

matching so tools that those tools that use this kind of algorithm are expected to have a wide applicability [Faria et al., 2014].

The substantial overlap between existing biomedical ontologies [Kocbek et al., 2012, Kamdar et al., 2015], makes ontology matching essential for integrating their information and ensuring interoperability between them [Faria et al., 2014]. There are various methods for finding these mappings, and they can be classified according to their granularity (entity-level vs. structural-level) or their interpretation of the input data (syntactic, external, or semantic) [Euzenat and Shvaiko, 2011].

In [Shvaiko and Euzenat, 2013] an analysis of the state of the art and future challenges in the ontology matching field is carried out. As evaluations of the recent years indicate, the field of ontology matching has made a measurable improvement, the speed of which, however, is slowing down. In order to achieve similar or better results in the forthcoming years, actions have to be taken. We believe this can be done through addressing specifically promising challenges that we identify as: (i) large-scale matching evaluation, (ii) efficiency of matching techniques, (iii) matching with background knowledge, (iv) matcher selection, combination and tuning, (v) user involvement, (vi) explanation of matching results, (vii) social and collaborative matching, (viii) alignment management: infrastructure and support.

2.4.4. Quality assurance in ontologies

Quality assurance has been addressed in several ways that require the combination of different activities at both textual and axiomatic levels. On the axiomatic side, the manual detection of irregularities like missing restrictions has been addressed in work such as [Rector et al., 2011, Rector and Iannone, 2012], whereas syntactic and semantic irregularities are detected by RIO [Mikroyannidi et al., 2011]. In addition, tools like OOPS are able to detect pitfalls in the axiomatisation of ontologies [Poveda-Villalón et al., 2012]. Hence, including methods that pinpoint anomalies would also help ontology developers to enrich their ontologies.

In the field of quality assurance the use of metrics is common practice in engineering activities, which also happens in ontology engineering. For example, metrics are widely used to evaluate ontology quality, correctness or similarity [Lozano-Tello and Gómez-Pérez, 2004, Tartir et al., 2005, García et al., 2010, Pesquita et al., 2009, Duque-Ramos et al., 2011].

Chapter 3

Objectives and methodology

In this chapter we motivate this thesis taking into account the analysis of the state of the art presented in Chapter 2.

3.1. Motivation

According to [Egaña-Aranguren, 2009], different aspects of biomedical ontologies might be desirable: rigour and axiomatic richness. The KRL chosen for representing an ontology helps to codify biomedical knowledge with rigour. KRLs, in turn, enable the use of different levels of axiomatisation. These levels range from the use of simple axioms like hierarchical relations to the use of more complex axioms. The more expressive the KR language is, the more complex axioms can be used to codified knowledge.

Often, rigour and axiomatic richness are independent aspects of biomedical ontologies. To what extent rigour and axiomatic richness are needed is difficult to measure. On the one hand, those biomedical ontologies used as simple plain taxonomies or controlled vocabulary do not need either rigour or complex axiomatisation. However, taxonomies were considered to be full ontologies [Studer et al., 1998, Gómez-Pérez, 1999]. On the other hand, those biomedical ontologies used as domain ontologies should be as rigorous and axiomatically rich as possible.

3.2. Main research question

Biomedical ontologies are released in public online repositories where both plain taxonomies and domain ontologies are mixed. This makes difficult both users and ontologies developers difference between them. Moreover, often the reason for this lack of rigour and axiomatisation is because biomedical ontology engineering has been more difficult for biologists than was expected [Yu, 2006, Ruttenberg et al., 2007], as it is explained in Figure 1.2 example. Using this as reference, our main research hypothesis is:

> *The axiomatic richness of biomedical ontologies might be improved by creating more complex axioms and using information that is codified in the hierarchy of concepts in a human friendly way but not machine exploitable.*

Consequently, this thesis comes up with a methodology that helps domain experts in the analysis and the enrichment of their ontologies. In the context of rigour and axiomatisation, we understand ontology enrichment as the capability of increasing the rigour or axiomatisation of an ontology. This motivation is also in one of the lines stated in [Stroetman et al., 2009] to achieve semantic interoperability in the medical domain: *"... the selected recommendations address actions focusing on content, tools and processes in the development of terminologies"* , being here terminologies understood as a sort of controlled vocabulary. The achievement of such semantic interoperability will depend, to some extent, on the usefulness of available biomedical ontologies. This motivation is also pushed by the results obtained in [Fernandez-Breis et al., 2010], where the enrichment of GO-MF was addressed. Our previous concerns triggered the next research question:

> *Could we develop an automatic method that could transform some of the content expressed in natural language in an ontology in logical axioms, and being the method systematically applicable to enrich biomedical knowledge resources?*

In our attempt to come up with an answer to this question this thesis progresses the topics that we have presented in Chapter 2. In summary, the more complete ontologies we have, the more interoperable the data will be. This will benefit many projects as data interoperability is also a key requirement for an efficient data analysis in translational medicine, by representing domain knowledge with ontologies [Machado et al., 2015].

3.3. Objectives

The main goal of this thesis is to analyse biomedical knowledge resources in order to support domain experts to identify *hidden semantics*, which can be converted into explicit formal content; this would contribute to the quality assurance of biomedical ontologies. To achieve this goal the following tasks and goals are defined:

- **O1.** Development and implementation of a methodology for the automatic characterisation of ontologies using the analysis of natural language associated with its concepts.

- **O2.** Development and implementation of a methodology for elucidating *hidden semantics* that can be used to generate axioms that contribute to the quality assurance biomedical ontologies.

  - **O2.1.** The methodology should be applicable to both small and large biomedical ontologies. This means it should be ready to scale regardless of the size and expressivity of source ontologies.
  - **O2.2.** The general methodology must be supported by a set of sub-methods that enable users to drive the study of the ontology from different semantics axes (e.g. semantic taxonomic relations, semantic associative relation, alignments and so on).
  - **O2.3.** The methodology should contextualise and relate *hidden semantics* within the hierarchy and links created though taxonomic and associative relations.
  - **O2.4.** The methodology should contextualise and relate *hidden semantics* within the context of other biomedical ontologies, so that this trigger the orthogonality and re-use principles for building good ontologies.
  - **O2.5.** The methodology should contribute to current methodologies for building ontologies like those presented in section 2.4.1.
  - **O2.6.** The methodology should be generic and systematically applicable to new versions or new biomedical ontologies.

- **O3.** Development and implementation of an integrated platform that helps users from a domain expert profile to use it.

    - **O3.1.** The platform should make complex ontological aspects as much transparent as possible.
    - **O3.2.** The use of the platform should avoid final users (domain experts with a lower technical profile) dealing with technical configuration problems.

- **O4.** Application and validation of the obtained results after applying the methodology to a set of relevant biomedical ontologies.

3.4. Methodology

During the development of this thesis we have iterated over the 4 main steps to be explained shortly. We have used the output and the lessons learned in step 4 to create new research hypotheses, refined some of the goals (detailing new sub-objectives), and in general contributing to the improvement of our base-line method.

1. Study of literature and state of the art:

    - Semantic Web: study of ontologies as a method for representing reality and in particular how ontologies are used in life sciences. We focused our attention in DL languages and its capabilities for reasoning, in particular we focused in OWL. We study how natural language can be found within ontologies and how it is related to the expressivity and semantics expressed in the ontology.
    - Bioinformatics: analysis of the literature for contextualising this thesis in the field of bioinformatics and how ontologies have contributed to the management and formalization of the information generated by this field.
    - Biomedical knowledge repositories: study of different sources of biomedical knowledge and how they could be improved by means of axiomatic enrichment and the benefits for translational medicine. In particular, we studied ontologies publicly available on the Internet through ontology repositories like BioPortal. Moreover, we have focused our interest in the study of two relevant ontologies

in the biomedical domain Gene Ontology and SNOMED CT, which remain to the biological and medical part in the context of biomedical ontologies.

- Ontology enrichment: study of the current methodologies and approaches related to ontology enrichment: ontology engineering, natural language processing techniques, automatic extraction of relations from texts, ontology learning, ontology matching and other work related to quality assurance. This study was focused on methods that use natural language content to generate axioms in the ontology.

2. Formalisation of the methods proposed in this thesis:

   - Development of a base-line method for the analysis of regularities in ontology labels. We formalise the definition of a *lexical regularity* and where they can be found within ontologies. As a first attempt of approach, we formalise the relation between *lexical regularity* and other elements within the ontology. We use this formalisation to analyse and to characterise biomedical ontologies according to their *lexical regularities.*

   - We formalise some concepts that let us measure different aspects using metrics. These metrics let users to rank *lexical regularities* according to different criteria. Here we use clustering techniques for classifying ontologies according to their adequacy to be enriched.

     - We develop a process to prioritise *lexical regularities* taking into account semantic aspect of the ontologies. In particular we formalise *lexical regularities* in the context of taxonomical relations.
     - We develop a process to formalise the relation between *lexical regularities* and content codified in other ontologies based on a state of the art work: cross-product extensions. This formalisation allowed our method to be compared to such a piece of work.

3. Development of the OntoEnrich platform, which permits us to run experiments using the proposed method and contribute to evaluating our research hypotheses.

   - The methods are implemented as a library so that they can be used and integrated with other solutions in the state of the art.

   - The methods are implemented using visualisation forms that help domain experts with low technical knowledge in the analysis and interpretation of the results.

4. Analysis of the obtained result and evaluation:

   - Due to the lack of gold standards against which to validate our results, we develop strategies that compare our solution to others that have been proposed in the state of the art. The analysis of this automatic comparison and a manual analysis of the results lets us to point out both advantages and downsides of our method.

   - We apply our method to different corpora of biomedical ontologies extracted from the web. In particular from BioPortal and the OBO Foundry repositories. Apart from this, we evaluate the method with the Gene Ontology and SNOMED CT.

Chapter 4

The OntoEnrich framework

This chapter contains a general overview of the OntoEnrich framework, which will be detailed in the publications shown in chapter 5. Additionally, we reference work published in peer review international conferences that have contributed to the development of this thesis. Therefore, the goals of this chapter are:

- Present the publications presented as part of this thesis.
- Justify these publications as a scientific unit.
- Unify some partial results.

4.1. General description of the method

First, it should be noted that strictly speaking, an ontology should not contain any instance, because it is supposed to be a conceptualisation of the domain. The combination of an ontology with associated instances is what is known as a *knowledge base* [Stevens et al., 2000]. This thesis is focused in the analysis of TBox but not in the ABox, so we focus our attention in those OWL constructors related with classes.

We used the work carried out in [Fernandez-Breis et al., 2010] as an initial methodological reference for the development of the thesis. Those results obtained in [Fernandez-Breis et al., 2010] showed that exploiting the *hidden semantics* within class labels offered significant benefits. The results were used for detecting patterns from a GO sub-hierarchy such as the following:

- (1) *"X binding"*: the selective, non-covalent, often stoichiometric interaction of a molecule with one or more specific sites on another molecule.
- (2) *"translation X factor activity"*: any molecular function that is involved in the initiation, activation, perpetuation, repression or termination of polypeptide synthesis at the ribosome.

These patterns inspired the core concept of this work: *lexical regularities*. The formal definition of *lexical regularity* can be found at [Quesada-Martínez et al., 2015d], a simplified definition is:

> *A lexical regularity is a group of consecutive ordered words that appear in more than one class of an ontology.*

In the previous examples, the *lexical regularities* are the fixed part of the patterns (e.g., binding, translation or factor activity). For example, *"binding"* appears in more than 1 600 labels in the GO Molecular Function ontology. Another example of a lexical regularity is the lexical regularity *"negative regulation"*, which in general stands for the prevention or reduction of a biological process. This linguistic expression appears in several biomedical ontologies, but it is not usually represented with logical axioms. The *"negative regulation of transcription"* and the *"negative regulation of translation"* in the Gene Regulation Ontology or the *"negative regulation"* in the Phenotypic Quality Ontology are similar examples. In this particular case, the text of the regularity can be aligned with an `owl:ObjectProperty` whose label is `'regulate'`. The majority of the activities described in [Fernandez-Breis et al., 2010] were performed manually, and ontology builders would require some support and some automation in order to make its application wider and more efficient. Based on the results in [Fernandez-Breis et al., 2010], we made our initial hypothesis that classes exhibiting *lexical regularities* may encode the meaning of a domain object, and there should be a relation between this class and other classes that exhibit that regularity.

During the development of this thesis that initial hypothesis was derived in the creation of the method shown in Figure 4.1. This figure shows the current stage of the method that we achieved after several iterations.

According to the state of the art, the method is contextualised in the field of *ontology engineering*, and in particular in *ontology enrichment*. The enrichment will be carried

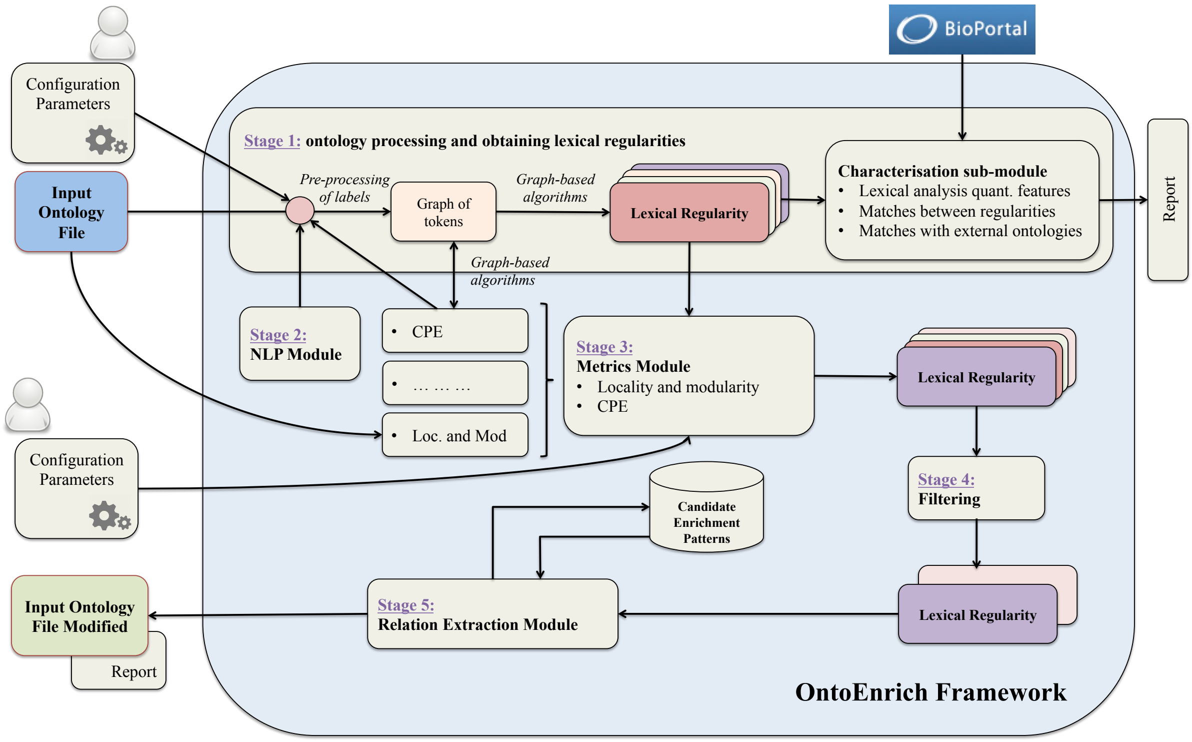


Figure 4.1: The OntoEnrich framework

out using information codified in the labels of the ontology not in external documents or other resources, for this reason this is out of the scope of *ontology learning*. Through the analysis of ontology labels we elucidate *hidden semantics* that let the user to enrich the ontology from an axiomatic point of view. This enrichment will make explicit knowledge so that *reasoner* can take advances of the *expressivity* offer by the *knowledge representation language* base on DL in which the ontologies were originally defined. The method takes an ontology as an input, and during stage 1 ontology labels are processed to build a graph that is used to calculate the whole set of *lexical regularities*. This process is performed according to some input configuration parameters. For example, different tokenisation strategies for the labels can be applied, since the use of blank as split character until the use of more advance text preprocessing techniques with the support of stage 2. Moreover, during this first stage *lexical regularities* are used to calculate some quantitative features. These features are used to lexically characterise the ontology according to its natural language identifiers generating a report. Some of these features make use of *ontology matching* to complete *lexical regularities* with content that is already defined in ontologies, this should promote the *re-use* of content in the biomedical community. Apart from this, in stage 3 we propose the use of metrics to study if *lexical regularities* can be used for generating new *taxonomical* or *associative relations*. For example, *semantic similarity* methods are also use to contextualise those classes that exhibit a *lexical regularity* taking into account the hierarchy already defined by *taxonomical relations*. Other alignment algorithms make use of the graph calculated the cross-product extension metric. For each metric a new configuration can be needed. These metrics enable their use in filtering methods (stage 4). Finally, those promising *lexical regularities* could be transformed into *ontology patterns* used to enrich the ontology (stage 5). As a result, the output would be the modified ontology plus a report about its changes. To sum up, the method proposed in this thesis contributes to scenarios 4, 5, 7 and 8 in the NeOn methodology for building and managing ontologies. Next, we further explain each stage.

Stage 1: ontology processing and obtaining lexical regularities

The formalisation of the method is presented in [Quesada-Martínez et al., 2015b], which is in section 5.1 of this document. There notions like *delimiter set*, *tokenise function*, *token*, *lexical regularity*, *sub/super regularity*, *exact/partial match* and other concepts toward the lexical analysis of an ontology are proposed. According to the survey about identifiers in ontologies mentioned in the state of the art, the method accepts annotations in ontology labels as well as processing the IRI fragments whether it is desirable.

Although the parameters of the algorithm are described in [Quesada-Martínez et al., 2015b], we briefly introduce two of them that let us continue describing the method: *coverage threshold* and *textual alignments*.

- *Coverage threshold*: the coverage threshold is the minimum percentage of classes in which a *lexical regularity* must appear to be included in the lexical analysis.

- *Textual alignments:* lexical alignments are found between *lexical regularities* and other elements within the ontology. The alignments also can be found in external ontologies so that contributes to the re-use of knowledge.

The *coverage threshold* is an input parameter of the algorithm. This plays an important role in the method. First, it offers users the possibility of selecting different levels of granularity based on the frequency of the *lexical regularities*. Second, this parameter is used to optimise the search of the regularities; and this is important with large ontologies like the Gene Ontology (around 65 000 classes) or SNOMED CT (more than 250 000 classes), being possible in SNOMED CT to have more than one label per class.

As the method must scale with large ontologies, it organises labels using a graph structure like the one shown in Figure 4.2. The graph shows the analysis of 4 class labels: (1) `‘positive regulation of isoprenoid’`, (2) `‘negative regulation of isoprenoid’`, (3) `‘vitamin binding’` and (4) `‘isoprenoid binding’`. Their lexical analysis yields a graph of 7 nodes (tokens) and highlights 4 shared tokens across the 4 labels. The token “regulation” is common in labels 1 and 2; thus, the corresponding node has two input arrows in the graph. Similarly, token “of” is shared across labels 1 and 2; thus, the incoming arrow of the corresponding node in Figure 4.2 has the label ids on the top. The direction of the arrow depicts the order of the tokens. For example, the *“regulation of isoprenoid”* regularity consists of 3 consecutive tokens that are used in labels 1 and 2. Similarly, *“binding”* is shared across labels 3 and 4.

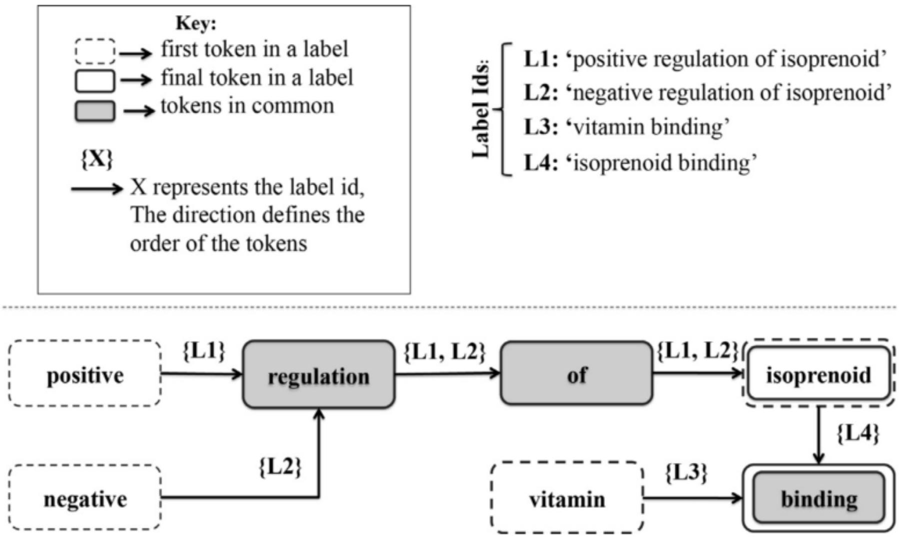
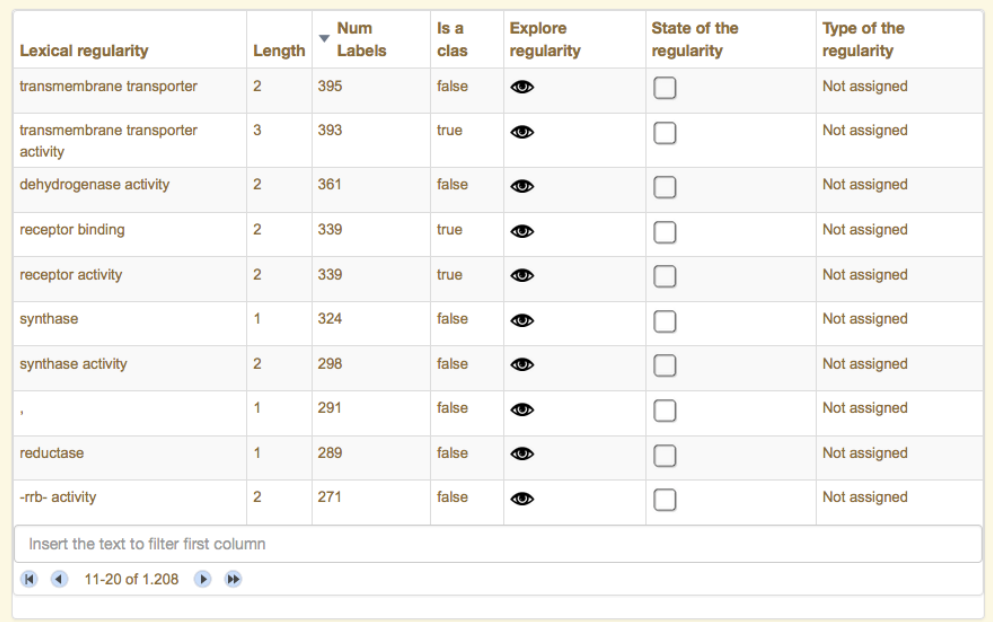


Figure 4.2: Graph of the content of 4 labels

Using this graph we are able to detect *simple* and *complex* identifiers like the one proposed by [Third, 2012]. Moreover, this graph is the central structure around which the method implements the *ontology alignment* algorithms or other NLP techniques like lemmatisation or nominalisation. An explanation of this graph is introduced in [Quesada-Martínez et al., 2015b]. However, the complete explanation of this graph including algorithms and a discussion that justifies its use can be found in [Quesada-Martínez et al., 2015d]. A complete analysis of the graph complexity can be found in Appendix A.



Lexical regularity	Length	Num Labels	Is a clas	Explore regularity	State of the regularity	Type of the regularity
transmembrane transporter	2	395	false	👁	☐	Not assigned
transmembrane transporter activity	3	393	true	👁	☐	Not assigned
dehydrogenase activity	2	361	false	👁	☐	Not assigned
receptor binding	2	339	true	👁	☐	Not assigned
receptor activity	2	339	true	👁	☐	Not assigned
synthase	1	324	false	👁	☐	Not assigned
synthase activity	2	298	false	👁	☐	Not assigned
,	1	291	false	👁	☐	Not assigned
reductase	1	289	false	👁	☐	Not assigned
-rrb- activity	2	271	false	👁	☐	Not assigned

Insert the text to filter first column

11-20 of 1.208

Figure 4.3: Visualisation of the list of *lexical regularities* and some basic features extracted from the Gene Ontology Molecular Function.

Using the proposed algorithms we are able to detect the whole set of *lexical regularities*, according to the *coverage threshold*. For example, Figure 4.3 shows 10 regularities obtained from GO MF. The GO MF version used has 8 547 labels, so if we use a *coverage threshold* of 0.1 % the number of lexical regularities is 1 208. As it can be seen in Figure 4.3, in the formalisation of the method some features associated with the *lexical regularities* are defined [Quesada-Martínez et al., 2015b]. In Figure 4.3 a couple of examples are shown: frequency (column "Num. Labels") or if the *lexical regularity* is a class in the ontology or not (column "Is a class"). This latter aspect is obtained by the lexical alignment algorithm. Then, as a result of the application of stage 1, the lexical characterisation of the ontology can be obtained.

Stage 2: Natural language processing module

Stage 1 makes use of one of the techniques proposed as part of a NLP pipeline, concretely tokenisation located in the *textual preprocessing* stage mentioned in [Ruiz-Martínez, 2011]. We have decided to include in the method different levels of tokenisation as the domain codified in the ontology could make the performance of the same method to be different according to the ontology used as input. The options are: (1) simple tokenisation based on white characters, (2) tokenisation, lemmatisation and part-of-speech using the Stanford NLP tokenisation trained with general English text corpora [Manning et al., 2014] and (3) nominalisation of verbs using the Specialist Lexicon [National, 2015]. Moreover, these NLP strategies are integrated in the graph's structure so that both the *lexical regularities* and alignment algorithms use them as a reference.

Stage 3: Metrics module

The basic features enable the ordination of the list of regularities. For example, higher frequencies might capture general patterns. Another indicator is to show whether the regularity corresponds to the name of a class. These and other features can be used in our approach to prioritise *lexical regularities* according to different semantic aspects that represent different types of *hidden semantics*. For example, the systematic naming captures taxonomical relations and the detection of regularities that are verbs can be exploited to generate other *associative relations*. For this reason, we propose to model different aspects related with a *lexical regularity* using a module based on metrics. Metrics are commonly used in practice in engineering activities, which also happens in ontology engineering. Moreover, as we commented in the state of the art metrics are widely used to evaluate ontology quality, correctness or similarity. In the scope of *lexical regularities*, the value of a metric depends on a function $f(x) = y$, where the domain of x is a *lexical regularity* and it range is a value where $y \in [m, n]$. The interpretation of y will depend on the the aspect/s measured by $f(x)$.

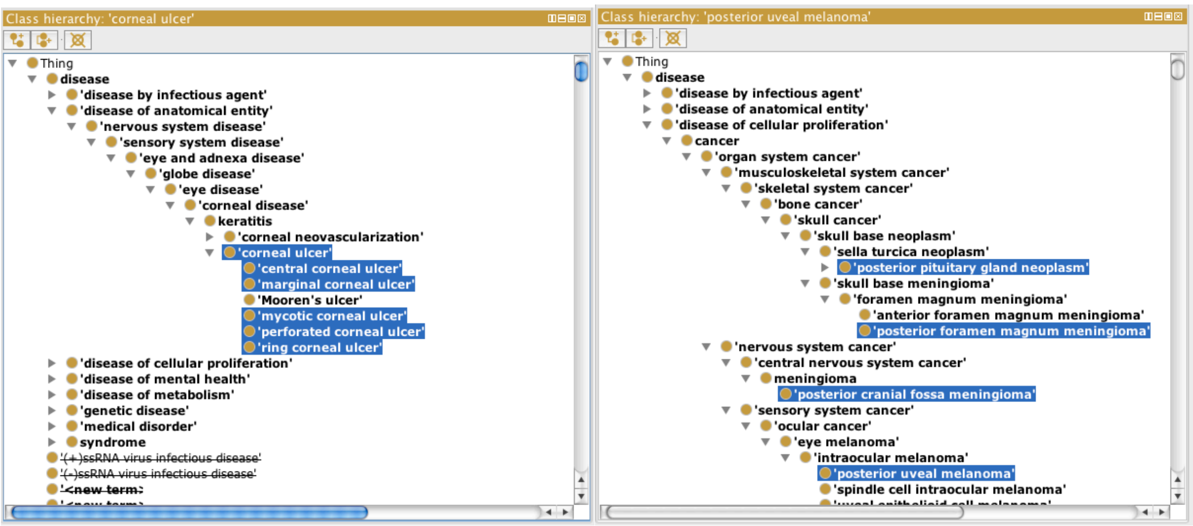
Stage 3.1: Modularity and locality metrics

In [Quesada-Martínez et al., 2014] we present a metric that measures the locality ($l(x)$) and modularity ($m(x, p)$) of those classes that exhibit a *lexical regularity*. These two aspects take into account the semantics codified in the hierarchy of concepts using taxonomical relations of the type `rdfs:subClassOf`. Figure 4.4 illustrates the two examples of the locality metric of regularitites *"colnear ulcer"* and *"posterior"* from the Human Disease Ontology [Schriml et al., 2012].

- Figure 4.4 (left) highlights those classes exhibiting the lexical regularity *"corneal ulcer"*. In this case, the ontology authors have followed a systematic naming in the class labeling that contributes to the lexical similarity. The class *"colnear ulcer"*, which matches the *lexical regularity*, is the ancestor of those classes that exhibit it, so the hierarchical relationships are already explicit. In this case, the locality measure of *"corneal ulcer"* is 0.85 (close to 1).
- Figure 4.4 (right) highlights those classes exhibiting the *lexical regularity "posterior"*. In this case, these classes do not appear in the same hierarchy so they do not codify taxonomical relationships. In this case, the locality measure of *"posterior"* is 0.14 (close to 0).

It should be pointed out that the low value of the *lexical regularity "posterior"* does not mean that this regularity is meaningless, but that those classes that exhibit it are not close in the asserted hierarchy defined in the ontology, which is the aspect measured by the locality function ($l(x)$). $l(x)$ makes use of semantic similarity measures that combine edge-based and graph-based approaches [Pesquita et al., 2009].

The locality metric is completed with the modularity metric ($m(x, p)$). $l(x)$ quantifies how close two classes that exhibit a given regularity are in the ontology. Hence, $m(x, p)$ may estimate how a regularity is distributed in a particular context of the ontology, which can be useful to identify modules in the ontology. For example, in the Gene Ontology the *lexical regularity "kinase activity"* is exhibited by classes like `'kinase activity'`, `'regulation of kinase activity'`, `'dolichol kinase activity'`, and `'protein histidine kinase activity'`. As we have explained in the introduction, ontologies often are organised in sub-modules that are represented by top-levels of the ontology. In the case of the Gene Ontology the



Class hierarchy: 'corneal ulcer'

- Thing
  - disease
    - 'disease by infectious agent'
    - 'disease of anatomical entity'
      - 'nervous system disease'
        - 'sensory system disease'
          - 'eye and adnexa disease'
            - 'globe disease'
              - 'eye disease'
                - 'corneal disease'
                  - keratitis
                    - 'corneal neovascularization'
                    - 'corneal ulcer'
                      - 'central corneal ulcer'
                      - 'marginal corneal ulcer'
                      - 'Mooren's ulcer'
                      - 'mycotic corneal ulcer'
                      - 'perforated corneal ulcer'
                      - 'ring corneal ulcer'
    - 'disease of cellular proliferation'
    - 'disease of mental health'
    - 'disease of metabolism'
    - 'genetic disease'
    - 'medical disorder'
    - syndrome
  - ~~'(+)ssRNA virus infectious disease'~~
  - ~~'(-)ssRNA virus infectious disease'~~
  - ~~'<new term>'~~
  - ~~'<new term>'~~

Class hierarchy: 'posterior uveal melanoma'

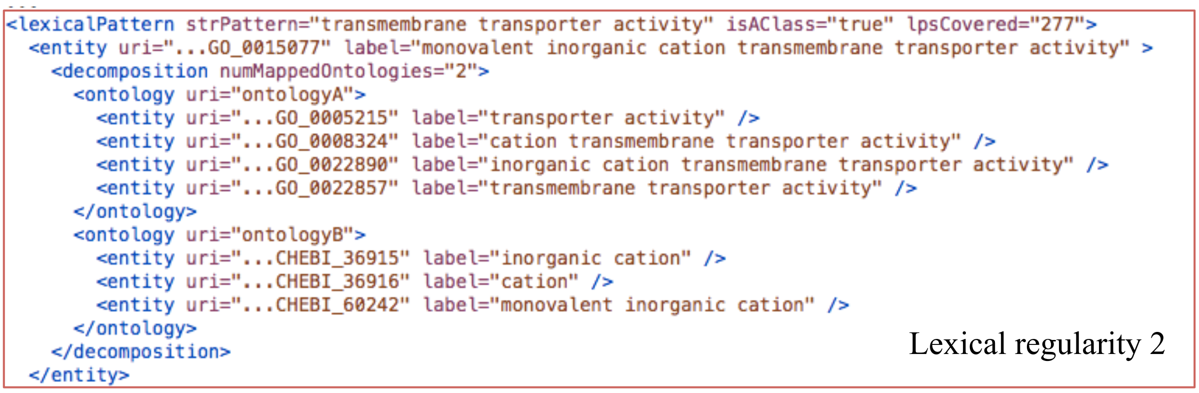
- Thing
  - disease
    - 'disease by infectious agent'
    - 'disease of anatomical entity'
    - 'disease of cellular proliferation'
      - cancer
        - 'organ system cancer'
          - 'musculoskeletal system cancer'
            - 'skeletal system cancer'
              - 'bone cancer'
                - 'skull cancer'
                  - 'skull base neoplasm'
                    - 'sella turcica neoplasm'
                      - 'posterior pituitary gland neoplasm'
                    - 'skull base meningioma'
                      - 'foramen magnum meningioma'
                        - 'anterior foramen magnum meningioma'
                        - 'posterior foramen magnum meningioma'
          - 'nervous system cancer'
            - 'central nervous system cancer'
              - meningioma
                - 'posterior cranial fossa meningioma'
            - 'sensory system cancer'
              - 'ocular cancer'
                - 'eye melanoma'
                  - 'intraocular melanoma'
                    - 'posterior uveal melanoma'
                    - 'spindle cell intraocular melanoma'
                    - 'uveal epithelioid cell melanoma'

Figure 4.4: Two examples of the hierarchy of the classes of the Human Disease Ontology. The highlighted classes are classes with a lexical regularity, "corneal ulcer" left hierarchy and "posterior" right hierarchy.

first level represents 3 different aspects with the classes: `'biological_process'` (BP), `'cellular_component'` (CC) and `'molecular_function'` (MF). Let us select these 3 classes as the context for calculating the modularity of the regularity. We select these classes as p. The regularity *"kinase activity"* appears in classes that are descendants of BP and MF, and corresponds to the full label of a class in MF. And this can be systematically applied to the whole set of *lexical regularities* using $m(x, p)$. Moreover, in BP, it is preceded by another *lexical regularity*, *"regulation of"*, as in `'regulation of kinase activity'`. As a consequence, the presence of these *lexical regularities* in two classes or more classes in p could help the ontology developer to make explicit the links between, in this example, the corresponding MF and BP classes.

Stage 3.2: Cross-product extension metrics (CPE)

In [Quesada-Martínez et al., 2015d] we present a metric that measures, following Third's terminology, if *lexical regularities* capture *complex identifiers* that can be defined as a composition of other *simple identifiers*, which are lexically contained in them. For example, let us to take again the *"X binding"* pattern from Figure 1.3. *"Binding"* is the *lexical regularity* and it is a class in the hierarchy. However, according to Firth´s notion, *"a word is characterised by the company it keeps"*, the analysis of those words that enfold the *lexical regularity* could provide complementary information. For example, 20 % of the classes that replace X in the binding example are found as classes in the Chemical Entities of Biological Interest ontology (ChEBI). The cross-product extension metric $cpe(x, O)$ measures this kind of situation. This metric is based on the ontology matching operation showed in Figure 2.12 and using a lexical matcher algorithm as reference. The input of this metric is a *lexical regularity*. In general, the alignment is performed between the classes that exhibit a regularity and the whole set of classes in O (O is an ontology); although as we explain in [Quesada-Martínez et al., 2015d] the same ontology used to extract the *lexical regularities* can play the role of O. Moreover, $cpe(x, O)$ measures the percentage of classes with *complex identifiers* that can be decomposed into smaller fragments. An example of the decomposition of the class `'monovalent inorganic cation transmembrane transporter activity'` is shown in Figure 4.5, being the regularity *"transmembrane transporter activity"*.



```
<lexicalPattern strPattern="transmembrane transporter activity" isAClass="true" lpsCovered="277">
  <entity uri="...GO_0015077" label="monovalent inorganic cation transmembrane transporter activity" >
    <decomposition numMappedOntologies="2">
      <ontology uri="ontologyA">
        <entity uri="...GO_0005215" label="transporter activity" />
        <entity uri="...GO_0008324" label="cation transmembrane transporter activity" />
        <entity uri="...GO_0022890" label="inorganic cation transmembrane transporter activity" />
        <entity uri="...GO_0022857" label="transmembrane transporter activity" />
      </ontology>
      <ontology uri="ontologyB">
        <entity uri="...CHEBI_36915" label="inorganic cation" />
        <entity uri="...CHEBI_36916" label="cation" />
        <entity uri="...CHEBI_60242" label="monovalent inorganic cation" />
      </ontology>
    </decomposition>
  </entity>
```

Figure 4.5: Fragment of an eXtensible Markup Language (XML) file with the alignments found between the class "monovalent inorganic cation transmembrane transporter activity" in GO and classes in CheBI.

Moreover, it is worth pointing out how we take advantage of the graph structure explained before for carrying out the alignment. Traditional lexical matcher algorithm searches lexical alignments comparing the identifiers of two labels. In our case, we compare tokens. For example, Figure 4.6 shows the partial alignment between the class `'ammonium ion metabolic process'`, which exhibits the regularity *"metabolic process"*, with classes in CheBI. The extraction of the graph above is the result of loading CheBI ontology in a graph. The extraction of the graph below is the result of loading GO ontology in another graph. The alignment is calculated by aligning parts of both graphs. In this example, we can see that sometimes overlap can appear as one *complex identifier* can be decomposed in more than one combination of classes.

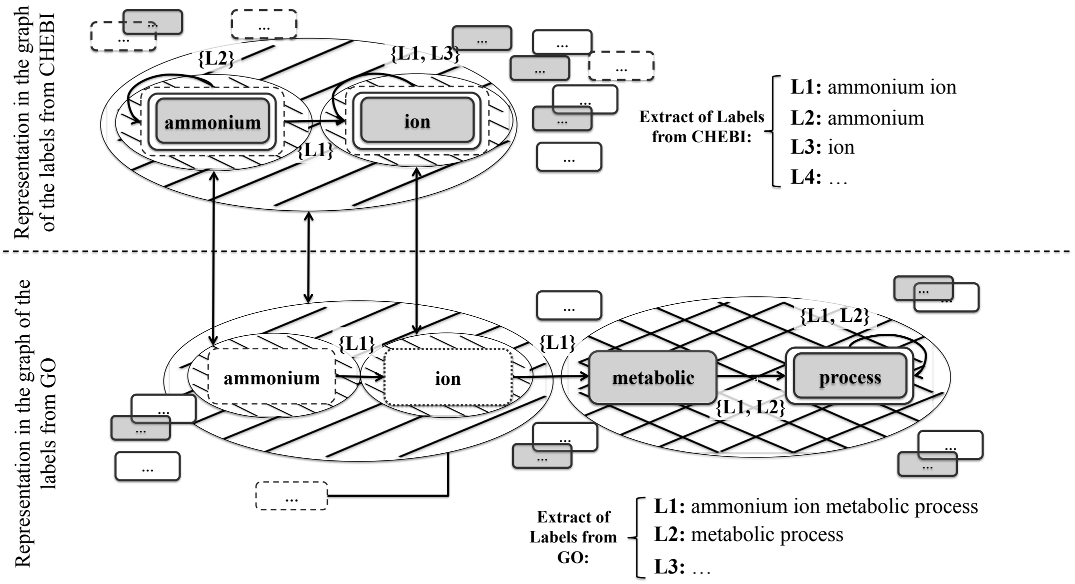

Representation in the graph of the labels from CHEBI
{L2}
{L1, L3}
ammonium
ion
{L1}
Extract of Labels from CHEBI:
L1: ammonium ion
L2: ammonium
L3: ion
L4: ...
Representation in the graph of the labels from GO
{L1}
{L1}
{L1, L2}
ammonium
ion
metabolic
process
{L1, L2}
Extract of Labels from GO:
L1: ammonium ion metabolic process
L2: metabolic process
L3: ...

Figure 4.6: Graphical representation (which represents the labels as graphs of tokens) of the decomposition of ammonium ion metabolic process using classes from GO and ChEBI. The graph would be formed by the whole set of labels from each ontology, but we show only some labels that participate in the decomposition of the class ammonium ion metabolic process.

Finally, we found a relation between this idea and the GO cross-product extensions [Mungall et al., 2011]. This is why we use the name *cpe* for this metric. The GO cross-product extensions provide logical definitions for GO classes using genus-differentia constructs of the form *"an X is a G that D"*. Here, *X* is the class that we are defining, *G* is the genus (more general class), and *D* is the *differentia*, a collection of characteristics that serves to discriminate instances of *X* from other instances of *G*. For example, the class '`mitochondrial translation`' can be seen as the genus '`translation`', and the *differentia* occurs inside a '`mitochondrion`'. In the context of our *lexical regularities* the hypothesis is that they could be capturing the genus of these cross-products, so we our *cpe metric* measures in such a situation.

Stage 4: Filtering

As we have mentioned before, *lexical regularities* can capture different types of *hidden semantics*. The *coverage threshold* implicitly lets the user apply filters based on the frequency of the *lexical regularities*. However, other filtering mechanisms would be desirable. On the one hand, we propose to use metrics values for applying filters according to the aspect defined by the metric. On the other hand, other aspects not interpretable by metrics should be included. For example, we add filters based on the part-of-speech tag labels associated with regularities.

Apart from these filters, we have detected certain overlaps as a consequence of the systematic naming convention in identifiers. These overlaps can be observed in the example of Figure 4.5. In our method, this situation is formalised through sub/super-regularity relations [Quesada-Martínez et al., 2015b]. For example, *"transporter"* is a regularity that is exhibited in 510 classes of GO MF. It has the super-regularity *"transporter activity"* (501) and *"transporter"* (9); this overlap can be observed in Figure 4.5 too. The method offers filters taking this situation into account [Quesada-martínez et al., 2013].

Stage 5: Relation extraction module

The desirable scenario is to automatise *lexical regularities* that are a result of stage 4 to automatically create axioms in the original ontology. As we have explained in the state of the art section, methods focus on the extraction of relation from texts and *ontology learning* [Hearst, 1992, Aussenac-Gilles and Jacques, 2008, Auger and Barrière, 2008b] make use of lexico-syntactic patterns. These patterns are predefined and used for search instances in text that fit to them. Each pattern has a semantical relation associated, so if patterns are found in a text the relation is directly created. Lexico-syntactic patterns take into account the part-of-speech analysis that indicates the syntactical category of text content. For example, the identifiers *"sphingolipid binding ."* and *"6-phosphogluconolactonase activity ."* follow the same lexico-syntactic pattern: adjective (JJ) followed by a noun (NN) and followed by a punctuation symbol (end of the label). Therefore, the enrichment pattern used for the first expression could be used for the second too. However, in the context of labels these types of patterns do not reveal the particular relation to create so they cannot be used to automatise the process. However, the experience gained with the application of methods like [Aussenac-Gilles and Jacques, 2008] suggested that verbs in text codify relations. So analysing *lexical regularities* that are verbs could be used, for example, to use that verb as `rdfs:objetcProperty`. Moreover, in the particular case of the GO Molecular Function, in patterns like *"X binding"* is the word "binding" that codifies the relation using an `rdfs:objetcProperty` labelled with `binds`, so is the nominalisation of the verb "binding" which hides the semantic relation.

So far, our method assists domain experts in the inspection of *lexical regularities* from different axes using metrics, but the automatic creation of the enrichment pattern has not been automatised, except for those cases like `'CX3C chemokine receptor binding'` and `'chemokine receptor binding'` where there is no reason that justifies the absence of a `rdfs:subClassOf` relation between them (Figure 2.8). In these cases, the systematic naming can be exploited to create an OPPL script that makes those classes that exhibit the super-regularity descendant of the sub-regularity, if the latter corresponds with the label of a class.

4.2. Results and Applications

This section includes a description of the result presented in the 3 publications that are part of this thesis. We organise this section as follows: for each experiment we describe the research hypothesis, the materials used and the results summary and some conclusions drawn.

4.2.1. Lexical characterisation of biomedical ontologies

- *Hypothesis:*
  - **H1:** the method can be scaled and systematically applied to biomedical ontologies. The formalisation of the method as the lexical characterisation of ontologies contributes to the next sub-hypothesis:
    - **H1.1.** Biomedical ontologies are rich in identifiers.
    - **H1.2.** Biomedical ontologies follow a systematic naming.
    - **H1.3.** Biomedical ontologies are in general plain taxonomies that have a low axiomatisation.
    - **H1.4.** Biomedical ontologies re-use content from other ontologies.

    - ○ **H1.5.** The lexical characterisation can help to select those ontologies that are appropriate to take advantages of methods that helps to elucidate *hidden semantics.*
  - **H2:** regularities are shared among ontologies. If a regularity is used to create an enrichment pattern the patterns could be applied to other ontologies.

- *Description:*
  - The method is systematically applied over a corpus of 178 OWL ontologies from the BioPortal repository.
  - We use the characterisation of ontologies based on quantitative metrics for classifying BioPortal ontologies according to their appropriateness to apply methods that detect *hidden semantics.*

- *Results:*
  - The formalisation and implementation of the method.
  - The lexical characterisation of ontologies, which includes information about:
    - ○ Characterisation of Axioms.
    - ○ Characterisation of *lexical regularities.*
    - ○ Re-Use of *lexical regularities.*
    - ○ Imported Ontologies.
    - ○ Distribution in cluster of the corpus of ontologies.

- *Conclusions:*
  - The ratio of identifier/classes is close to 1, which means that the ontology builders provide a textual definition. We find that 41 ontologies do not define labels but their identifiers contain text content. The use of labels instead of codifying the identifier in the URI is gaining popularity, as 65 % of ontologies chose this option. In a previous survey [Nor Azlinayati Abdul et al., 2010] the majority of ontology developers codified identifiers in the URIs. These results confirm hypothesis **H1.1**.

- Biomedical ontologies are rich in *lexical regularities*. Using a *coverage threshold* of 1 %, we obtain 8175 *lexical regularities* in 118 biomedical ontologies. This is an indicator that biomedical ontologies use a systematic naming and confirm **H1.2**. This also confirms the scalability of the method to the whole repository.

- 75 % of the axioms of BioPortal ontologies are annotations properties or `rdfs:subClassOf`, which we interpret as showing that most BioPortal ontologies are taxonomies and are rich in natural language content. This confirm hypotheses **H1.1** and **H1.3**.

- The majority of the BioPortal ontologies do not re-use concepts from other ontologies through imports, so detecting common *lexical regularities* could potentially re-use entities. 77.22 % of the ontologies do not import any other ontology. This partially rejects **H.1.4**. This 77.22 % of the ontologies that do not import ontologies could benefit from our method for finding regularities that appear in external ontologies. However, 15.60 % of the *lexical regularities* correspond to full labels of classes in the ontologies; that is, class labels are *complex identifiers* including labels from other classes. In addition, 36.44 % of these labels correspond to classes from external ontologies. These results suggest that the number of links between ontologies is lower than the degree of potential relation between the content of different ontologies. Hence, **H1.4** can be rejected. The fact that most alignments have been found in external ontologies, suggests that the re-use of this content for enriching current biomedical ontologies is a significant contribution. It should be pointed out that our approach measures the re-use in terms of explicit imports, which could be complemented with the study of use of URIs.

- An ontology is potentially suitable for enrichment whether its labels contain regularities, such regularities are exhibited by many classes in the ontology, and the regularities have matches with content from external ontologies. The cluster analysis reveals 3 clusters. Cluster 1 (42 members) and Cluster 3 (33 members) contain ontologies with such properties, so 75 % of the BioPortal ontologies analysed could benefit from enrichment processes. Cluster 2 contains 24 ontologies. This confirms **H1.5**.

- We found that 23.49 % of the regularities appear in more than one ontology.

This suggests that the axiomatisation using these regularities as a base to define an enriching pattern would contribute to their re-use between ontologies. This confirms **H2**.

4.2.2. Lexical regularities and taxonomic relations

- *Hypothesis:*
  - **H1** the locality of the regularities gives information useful for driving the axiomatic enrichment of the ontology.
- *Description:*
  - The method is extended to the locality and modularity metrics. These metrics use semantic similarly metrics between the set of classes that exhibit a *lexical regularity.*
  - The lexical analysis and the prioritisation of the *lexical regularities* is performed over four biomedical ontologies, which were selected due to their size and different content: 1) the Human Disease Ontology; 2) the Chemical Entities of Biological Interest; 3) the Gene Ontology; and 4) SNOMED CT . We perform their lexical analysis with different *coverage thresholds* and:
    - Calculate the locality metric value for all the *lexical regularities* obtained. In summary, 0 means that those classes that exhibit a *lexical regularity* are far in the hierarchy and 1 the opposite.
    - Calculate the modularity metric value using as *classes of interest* those that are in the first level of ontology classes. If the value of the metric counts the number of *class of interest* that are ancestors of the classes that exhibit the *lexical regularities.*
- *Results:*
  - The formalisation of the metric module, the locality and modularity metrics.
  - Integration of the metric module into our general method.
  - Implementation of the metrics.

- *Conclusions:*

    - Locality and modularity contribute to a better understanding of the engineering of the ontologies and may support domain experts in the prioritising of the most promising parts of the ontologies for axiomatic enrichment.

        - The mean value of the locality measure ranges from 0.20- 0.48, which means that, on average, *lexical regularities* are distributed along the hierarchy.
        - The mean percentage of *classes of interest* in which a *lexical regularity* appears is 52.79 % (1.56) in the ChEBI, 31.62 % (2.48) in the DOID, 67.06 % (2.0) in the GO, and 28.07 % (5.32) in SNOMED CT (the figures in brackets are the absolute values, for example, *lexical regularities* from the GO appear as descendant of 1.56 *classes of interest*). The absolute values of the modularisation measure reveal that, on average, those classes that exhibit a *lexical regularity* appear in more than one *class of interest*, which is a sign of the potential links between the lexical entities of different modules. Although the refinement of the *classes of interest*, for example, including classes in other levels will provided further details in order to enrich the ontology.

    These results confirm **H1**, although these metrics require human intervention for configuring the *classes of interest.*

4.2.3. GO MF analysis

- *Hypothesis:*

    - **H1:** the initial hypothesis is that *lexical regularities* by themselves can help the domain expert to automatically detect those enrichment patterns identified in [Fernandez-Breis et al., 2010].
    - **H2:** the inspection of sub/super-regularities would contribute to discover more specific patterns. The *lexical regularities* can be automatically converted into OPPL scripts that enrich the ontology.

- *Description:*

- We applied the method for detecting *lexical regularities* and characterising ontologies according to some quantitative features. The method includes a naive alignment algorithm that uses BioPortal search web services [Whetzel et al., 2011b], as well as alignments between regularities found in different ontologies.
- We applied the method to the same ontology used and enriched using knowledge patterns in [Fernandez-Breis et al., 2010].

▪ *Results:*

- The implementation of the method
- GO MF has 8 547 labels, which contain 36 944 tokens (6 808 unique tokens). The longest label has 29 tokens. 5 968 *lexical regularities* have been found using our tool. The longest pattern contains 22 tokens, the mean length is 2.7280, and the median is 2. Concerning the number of repetitions of each regularity, the mean value is 8, and the median is 3. 7.1 % of the *lexical regularities* correspond to the exact labels of GO MF classes, whereas 38.82 % correspond to exact labels of BioPortal ontology classes.

▪ *Conclusions:*

- The analysis of the binding taxonomy reveals that *"receptor activity"*, *"codon-amino-acid activity"*, *"hormone receptor activity"* and *"modification guide activity"* are the most frequent *lexical regularities*. Most of such patterns were manually identified in [Fernandez-Breis et al., 2010] by using knowledge patterns. The expert read the *knowledge patterns* and searched classes in the ontologies that follow a *lexical regularity* that capture the *knowledge patterns*. However, *lexical regularities* like *"hormone receptor activity"* were not considered in that effort because no specific *knowledge pattern* was defined for it, which shows the goodness of having such tooling support and contributes to confirm **H2**.
- The enrichment patterns proposed in [Fernandez-Breis et al., 2010] are a subset of the lexical regularities. Although the support of the method helps users

to elucidate *hidden semantics* the process for creating patterns to enrich the ontology still requires manual intervention. **H1** can be rejected.

4.2.4. Gene Ontology cross-product extension

- *Hypothesis:*
    - **H1:** the CPE-metric, and in particular the 3 conditions that we propose, provides information about the degree and type of enrichment that can be expected by analysing the content surrounding the text that is repeated in the *lexical regularity.*
    - **H2:** the classes captured by the *lexical regularities* can be used to enrich the ontology, in a similar way as with cross-products.
    - **H3:** the alignment method based on parts of labels provides more information than using alignments between the whole label.
- *Description:*
    - We performed lexical analysis on Gene Ontology for several reasons:
        - GO provides a controlled vocabulary for the functional annotation of gene products. To date, GO classes have been used to produce millions of annotations. Its enrichment would have an impact on the exploitation possibilities of the GO.
        - Our analysis of BioPortal ontologies revealed the *prima facie* suitability of the GO for its enrichment: 100 % of the classes have labels, 92 % of the words of the labels are repeated, and 85 % of the ontology labels exhibited 67 *lexical regularities.*
        - The GO consortium and other scientists have already identified the need of increasing the axiomatic richness of GO, and have recently developed a partially enriched version, the GO cross-product extensions.
    - We studied and described the similarities between *lexical regularities* and cross-products. We modelled these relations using the CPE-Metric.

    - The CPE-class condition allows for filtering classes that are based on exact textual alignments. In other words, they are based on an estimation of the enrichment of the *lexical regularities*; for this we include information from an external ontology and find decompositions of labels using tokens as the minimal representational unit.

  - Design and implementation of a validation strategy based on cross-products, which let us compare the performance of our method against a reference method.

    - For the comparison, we defined a template that measures the equivalence between our method and the reference method. Despite the previous work not being a gold standard, the fact that both methods share the same objective of the axiomatic enrichment of the GO, their expertise in the biological domain and the process followed (including manual curation), makes the reference method relevant for the evaluation of our results. The use of this template let us discuss our method in terms of the standard metrics of precision, recall and F1-measure using.

- *Results:*

  - Formalisation and implementation of the cpe-metric.
  - Integration of the cpe-metric in the general method.
  - The label of the classes of the GO are highly regular in lexical terms, and the exact matches with labels of external ontologies affect 80 % of the GO classes.
  - The CPE metric reveals that 31.48 % of the classes that exhibit regularities have fragments that are classes into two external ontologies that are selected for our experiment, namely, the Cell Ontology and the Chemical Entities of Biological Interest ontology, and 18.90 % of them are fully decomposable into smaller parts.

- *Conclusions:*

  - The CPE metric permits our method to detect GO cross-product extensions with a mean recall of 62 % and a mean precision of 28 %. This partially confirms **H2**. However, the overlap is not complete so the study is completed with an analysis of false positives to explain this precision value.

- CPE-c1 and CPE-c2 provide information about a greater number of general decompositions than those provided by the reference method and those proposed by the CPE-c3. This information is useful for domain experts but not for the automation process. This accepts **H1**. This fact also support **H3**, however as mentioned the information can introduce noise so filtering methods would be required.

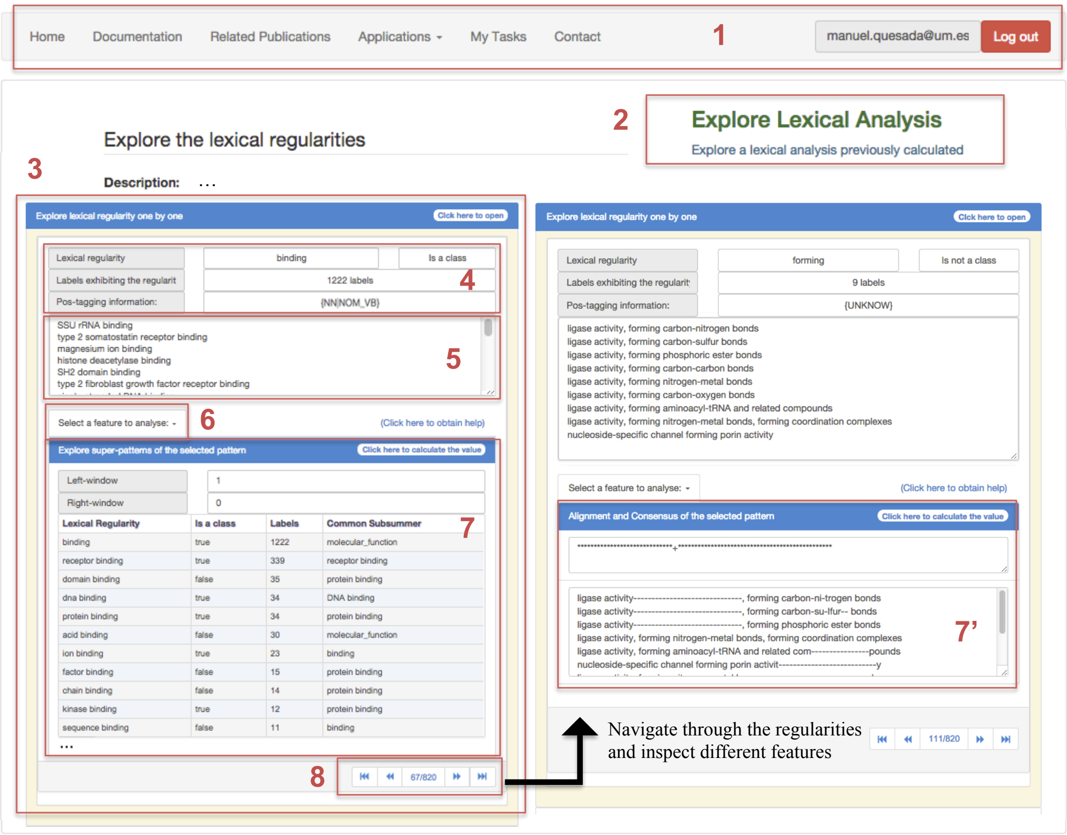
4.3. OntoEnrich tool

All the methods proposed in this thesis have been implemented and are available at:

`http://sele.inf.um.es/ontoenrich/`

The implementation [Quesada-Martínez et al., 2015c] is an online tool that avoids domain experts with low technical knowledge dealing with configuration or performance problems. The platform requires users to be logged in and to manage jobs using a task schedule. Due to the time use for executing some algorithms that implement the method, a task system lets users apply different stages without having to wait in front of the computer. Figure 4.7 shows screenshots for the *"binding"* and *"forming" lexical regularities* (LRs). This form is focused in a particular regularity but a more general inspection can be done using a form that contains a table like the one shown in Figure 4.3. The calculation of metrics can be done systematically for the whole set of regularities. When the task has finished the metrics values for all the regularities can be dynamically added as a new column to the table in Figure 4.3.

Coming back to the particular example of Figure 4.7, panel number 3 shows the information of the LR under inspection. We can navigate through the LRs (see Figure 4.7- 8). In Figure 4.7- 4 the general descriptors of the active LR are shown, and the labels that exhibit the LR can be explored in Figure 4.7- 5. More complex features of the LR are analysed independently and they are chosen using Figure 4.7- 6. Panel 5 shows the labels in which the LR appears. Panel 7 contains information about the super-patterns, sub-patterns, or alignment of labels, depending on the option selected in Panel 6.



Explore Lexical Analysis

Explore a lexical analysis previously calculated

Explore the lexical regularities

Description: . . .

Explore lexical regularity one by one

Lexical regularity	binding	Is a class
Labels exhibiting the regularit	1222 labels	
Pos-tagging information:	{NN|NOM_VB}	

SSU rRNA binding
type 2 somatostatin receptor binding
magnesium ion binding
histone deacetylase binding
SH2 domain binding
type 2 fibroblast growth factor receptor binding

Select a feature to analyse: ▾

Explore super-patterns of the selected pattern

Left-window	1
Right-window	0

Lexical Regularity	Is a class	Labels	Common Subsummer
binding	true	1222	molecular_function
receptor binding	true	339	receptor binding
domain binding	false	35	protein binding
dna binding	true	34	DNA binding
protein binding	true	34	protein binding
acid binding	false	30	molecular_function
ion binding	true	23	binding
factor binding	false	15	protein binding
chain binding	false	14	protein binding
kinase binding	true	12	protein binding
sequence binding	false	11	binding

. . .

Explore lexical regularity one by one

Lexical regularity	forming	Is not a class
Labels exhibiting the regularit	9 labels	
Pos-tagging information:	{UNKNOW}	

ligase activity, forming carbon-nitrogen bonds
ligase activity, forming carbon-sulfur bonds
ligase activity, forming phosphoric ester bonds
ligase activity, forming carbon-carbon bonds
ligase activity, forming nitrogen-metal bonds
ligase activity, forming carbon-oxygen bonds
ligase activity, forming aminoacyl-tRNA and related compounds
ligase activity, forming nitrogen-metal bonds, forming coordination complexes
nucleoside-specific channel forming porin activity

Select a feature to analyse: ▾

Alignment and Consensus of the selected pattern

```
*****************************+***********************************************
```

```
ligase activity-----------------------------, forming carbon-ni-trogen bonds
ligase activity-----------------------------, forming carbon-su-lfur-- bonds
ligase activity-----------------------------, forming phosphoric ester bonds
ligase activity, forming nitrogen-metal bonds, forming coordination complexes
ligase activity, forming aminoacyl-tRNA and related com----------------pounds
nucleoside-specific channel forming porin activit--------------------------y
```

Navigate through the regularities and inspect different features

Figure 4.7: Example of the online inspection of lexical regularities (`http://sele.inf.um.es/ontoenrich/files/ekaw2014ontoenrichImg.pdf`

- **Use Case 1 - “binding” (Figure 4.7 left):** this LR is quite general, so the inspection of the super-regularities can be useful. For example, there are 23 classes that exhibit the super-regularity *“ion binding”*, which is a class in the ontology; however, the least common subsumer of these 23 classes is `‘binding’` instead of `‘ion binding’`, which suggests the inspection of the labels that exhibit *“ion binding”* for discarding the irregularities in the naming of the labels. Hence, this analysis could serve to inspect the correlation between the *lexical regularities* and relationships between the corresponding classes.

- **Use Case 2 -"forming" (Figure 4.7 right):** this LR is recognised as a verb by the NLP modules. If we align and analyse the labels that exhibit this LR, the first 6 labels could be generalised as: *'ligase activity, forming ?x'*. Then, if ?*y* represents classes that follow such a pattern, these classes can be enriched with the axioms `'?y subClassOf 'ligase activity'` and `'?y subClassOf enables some (forming some ?x)'`, where the LR is created as an object property. However, the alignment of labels that exhibit the LR does not obtain a consensus as `'nucleoside-specific channel forming porin activity'` does not follow the pattern *"Y, forming X"*. In the other 2 labels, several elements are formed, so two axioms with an AND clause might be created.

Chapter 5

Publications composing the PhD Thesis

5.1. Lexical Characterization of Bio-ontologies by the Inspection of Regularities in Labels

Title	Lexical Characterisation of Bio-Ontologies by the Inspection of Regularities in Labels
Authors	Manuel Quesada-Martínez
	Jesualdo Tomás Fernández-Breis
	Robert Stevens
Pages	165-176
Volumen/Issue	10(2)
Type	Journal article
Journal	Current Bioinformatics
Impact factor (2014)	0,921
Publisher	Bentham Science
Year	2015
ISSN	1574-8936(Print) - 2212-392X (Online)
DOI	10.2174/157489361002150518124739
URL	http://benthamscience.com/journals/current-bioinformatics/volume/10/issue/2/page/165/
State	Published

Abstract: Hundreds of biomedical ontologies have been produced, with many of the significant, widely used ones being developed in collaborative efforts and following a set of construction principles, which include using a systematic naming convention for their labels. Despite their success, many of these ontologies have lacked a foundation of axioms that would expose the wealth of knowledge in the ontologies to computational reasoning. Our previous results suggest that exploiting the structure of the labels may contribute to

an axiomatic enrichment. Hence, in this work we perform a study of the structure of the labels of the ontologies available in BioPortal to classify them in terms of potential interest for axiomatic enrichment.

5.2. Prioritising Lexical Patterns to Increase Axiomatisation in Biomedical Ontologies

Title	Prioritising Lexical Patterns to Increase Axiomatisation in Biomedical Ontologies
Authors	Manuel Quesada-Martínez
	Jesualdo Tomás Fernández-Breis
	Robert Stevens
	Eleni Mikroyannidi
Pages	56-64
Volumen/Issue	(Vol. 54): Issue 1 2015
Type	Journal article
Journal	Methods of Information in Medicine
Impact factor (2014)	2,248
Publisher	Schattauer Publishers, Stuttgart
Year	2015
ISSN	0026-1270
DOI	http://dx.doi.org/10.3414/ME13-02-0026
URL	http://methods.schattauer.de/en/contents/archivestandard/issue/2168/manuscript/21370.html
State	Published

Abstract:

Introduction: This article is part of the Focus Theme of Methods of Information in Medicine on “Managing Interoperability and Complexity in Health Systems”.

Objectives: In previous work, we have defined methods for the extraction of lexical patterns from labels as an initial step towards semi-automatic ontology enrichment methods. Our previous findings revealed that many biomedical ontologies could benefit from enrichment methods using lexical patterns as a starting point.Here, we aim to identify which lexical patterns are appropriate for ontology enrichment, driving its analysis by metrics to prioritised the patterns.

Methods: We propose metrics for suggesting which lexical regularities should be the starting point to enrich complex ontologies. Our method determines the relevance of a lexical pattern by measuring its locality in the ontology, that is, the distance between the classes associated with the pattern, and the distribution of the pattern in a certain module of the ontology. The methods have been applied to four significant biomedical ontologies including the Gene Ontology and SNOMED CT.

Results: The metrics provide information about the engineering of the ontologies and the relevance of the patterns. Our method enables the suggestion of links between classes that are not made explicit in the ontology. We propose a prioritisation of the lexical patterns found in the analysed ontologies.

Conclusions: The locality and distribution of lexical patterns offer insights into the further engineering of the ontology. Developers can use this information to improve the axiomatisation of their ontologies.

5.3. Approaching the axiomatic enrichment of the Gene Ontology from a lexical perspective

Title	Approaching the axiomatic enrichment of the Gene Ontology from a lexical perspective
Authors	Manuel Quesada-Martínez
	Eleni Mikroyannidi
	Jesualdo Tomás Fernández-Breis
	Robert Stevens
Pages	35–48
Volumen/Issue	Volume 65, Issue, 1
Type	Journal article
Journal	Artificial Intelligence in Medicine
Impact factor (2014)	2,019
Publisher	Elsevier
Year	2015
ISSN	0933-3657
DOI	http://dx.doi.org/10.1016/j.artmed.2014.09.003
URL	http://www.aiimjournal.com/article/S0933-3657 %2814 %2900123-7/abstract
State	Published

Abstract:

Objective: The main goal of this work is to measure how lexical regularities in biomedical ontology labels can be used for the automatic creation of formal relationships between classes, and to evaluate the results of applying our approach to the Gene Ontology (GO).

Methods: In recent years, we have developed a method for the lexical analysis of regularities in biomedical ontology labels, and we showed that the labels can present a high degree of regularity. In this work, we extend our method with a cross-products extension (CPE) metric, which estimates the potential interest of a specific regularity for axiomatic enrichment in the lexical analysis, using information on exact matches in external ontologies. The GO consortium recently enriched the GO by using so-called cross-product extensions. Cross-products are generated by establishing axioms that relate a given GO class with classes from the GO or other biomedical ontologies. We apply our method to the GO and study how its lexical analysis can identify and reconstruct the cross-products that are defined by the GO consortium.

Results: The label of the classes of the GO are highly regular in lexical terms, and the exact matches with labels of external ontologies affect 80 % of the GO classes. The CPE metric reveals that 31.48 % of the classes that exhibit regularities have fragments that are classes into two external ontologies that are selected for our experiment, namely, the Cell Ontology and the Chemical Entities of Biological Interest ontology, and 18.90 % of them are fully decomposable into smaller parts. Our results show that the CPE metric permits our method to detect GO cross-product extensions with a mean recall of 62 % and a mean precision of 28 %. The study is completed with an analysis of false positives to explain this precision value.

Conclusions: We think that our results support the claim that our lexical approach can contribute to the axiomatic enrichment of biomedical ontologies and that it can provide new insights into the engineering of biomedical ontologies.

Chapter 6

Conclusions and future work

6.1. Contributions

The proposed method enables the inspection of *lexical regularities* in biomedical ontology labels. This analysis helps users to elucidate *hidden semantics* that can trigger the development of new logical axioms, which enables applications using biomedical ontologies to take real advantage of the expressivity capabilities of knowledge representation languages like OWL DL. The main contributions of this thesis are:

- The methodology for analysing ontologies based on *lexical regularities* in class labels.
- The scalable implementation of the method, due to the following features:
    - The graph organisation for labels, which speeds up the process of searching *lexical regularities* and used the *coverage threshold* as a mechanism for optimising and pruning the search.
    - The metrics, which prioritise *lexical regularities* according to different aspects related to properties of ontologies like semantic distance, modularity and/or alignments based on textual similarity.
    - The graph structure lets us implement an ontology matching alignment algorithm based on partial alignments instead of the whole label. Optionally,

the graph uses pre-processing techniques for obtaining the tokens like lemmatisation, pos-tagging or nominalisation. In this case, the alignment takes advantage of them as well.

- The application of the method to a number of biomedical repositories in order to:
    - Characterise BioPortal ontologies based on the content codified in their labels and matches between *lexical regularities* and other ontologies. We used the method to create clusters of ontologies according to their adequacy to be used in enrichment methods.
    - Application of the method to the Gene Ontology and study how the lexical analysis reconstructs the cross-products previously addressed by the Gene Ontology Consortium. This helps us to validate the method against previous work where relations were created.
- The availability of a web application for performing lexical analysis and exploring the *lexical regularities*. The visualisation of the *lexical regularities* using different semantic dimensions that helps domain experts to elucidate and analysis *hidden semantics*.

Therefore, our contribution helps to the automatisation of detecting *lexical regularities* that trigger the development of knowledge patterns which migth be transformed in Ontology Design Patterns (ODPs) to enrich the ontology.

6.2. Research questions

Next, we discuss the research hypotheses defined for this thesis, which were introduced in section 3.2:

- *The axiomatic richness of biomedical ontologies might be improved by creating more complex axioms and using information that is codified in the hierarchy of concepts in a human friendly way but not machine exploitable.*

  Biomedical ontologies are rich in identifiers. The ratio of identifiers/classes in 178 ontologies from BioPortal is close to 1. Ontology developers follow a systematic naming. As a consequence, 8 175 *lexical regularities* were obtained by our method with a *coverage threshold* of 1 % in the mentioned corpus. Comparison of the types of axioms revealed that 75 % of axioms were annotations properties or `rdfs:subClassOf`, which is an indicator that many of such ontologies are plain taxonomies or controlled vocabularies. Moreover, we found that 15.60 % of the *lexical regularities* correspond to full labels of other classes, and 36.44 % of these matches remain external ontologies suggesting that these lexical matches could be made machine exploitable by creating links between the classes that exhibit the *lexical regularities* and the classes matched. This makes more sense because 77.22 % of the ontologies in our corpus do not import other ontologies. These data support our initial hypothesis that biomedical ontologies can be enriched using information, which is already codified in a human friendly but not as axioms, so the application of enrichment methods is valuable. Moreover, our method lets us classify them according to their adequacy for their enrichment using their lexical characterisation.

- *Could we develop an automatic method that transforms some of the content expressed in natural language in an ontology in logical axioms, and being the method systematically applicable to enrich biomedical knowledge resources?*

The method enables the automatic detection and inspection of *lexical regularities* in biomedical ontologies identifiers from different axes. Moreover, 23.40 % of the *lexical regularities* appear in more than one ontology. This suggests that the axiomatisation using them as a base to define the enriching pattern would contribute to systematic application of the method among different biomedical ontologies.

The experiment was carried out with 5 large and relevant biomedical ontologies and has revealed that the classes that exhibit a *lexical regularity* are distributed along the hierarchy, according to the locality value, which ranges from 0.2- 0.48, and the modularity distribution (on average those classes that exhibit a *lexical regularity* are descendant of more than one *classes of interest*). These values, together with the information about matches, are an indicator of the type of knowledge that is captured by *lexical regularities*, and is a sign of the potential links. However, so far the automatic transformation of *lexical regularities* into ODPs has not been addressed beyond taxonomical relations of the type `rdfs:subClassOf`. Although we cannot automatically set the relations, we compare the classes captured by *lexical regularities* to those that were used to enrich the Gene Ontology using cross-products. This is modelled using the cpe-metric obtaining mean recall of 62 % and a mean precision of 28 %.

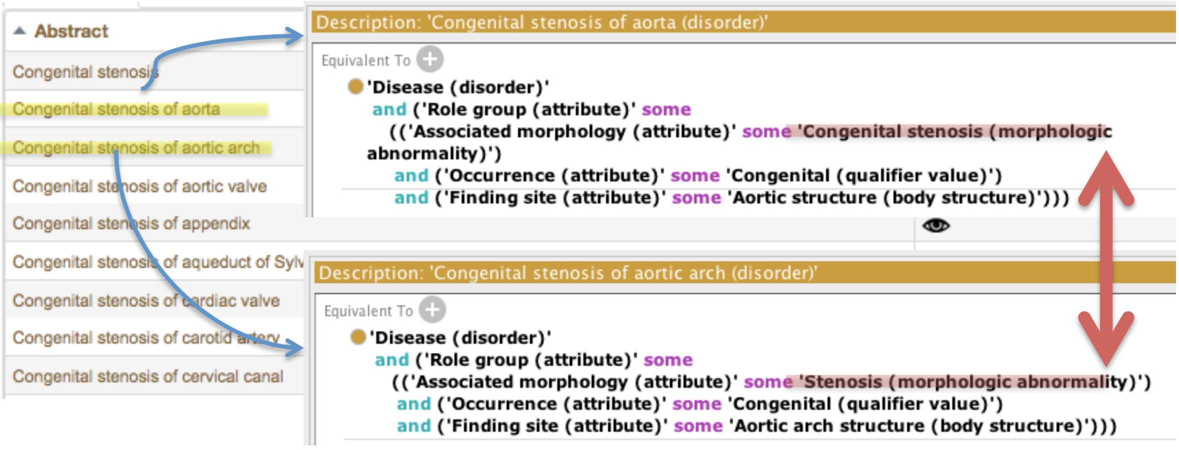
In summary, the method contributes to the automatic and systematic analysis of biomedical ontologies. However, the experiments using the two metrics presented have revealed that the *hidden semantics* behind a *lexical regularity* can be different. While sub/super-regularities that are classes can be used to create a hierarchical relation and this can be quantified with the locality metrics, *lexical regularities* that are verbs can be used to create other types of associative relations. These differences must be taken into account to automatically transform *lexical regularities* into patterns that enrich the ontology, so being able to improve this part becomes part of future work.

6.3. Brief discussion of future work

In this section we include some discussion that extends that of the papers and introduce lines of future work.

- We have manually studied patterns used for extracting semantical relations from texts. Patterns like *"X like Y, Z, T"* do not match with the content in ontology labels. However, in a preliminary study we have applied the algorithm for detecting *lexical regularities* but using as input the tags of the tokens obtained by pos-tagging techniques. As it was expected the most frequent patterns were formed by nouns, so the automatic extraction of the relations is complex. However, we have found some verbs or nominalisation of verbs. For example, as we have commented in *"vitamin binding"*, *"binding"* is the nominalisation playing the nominalisation of the verb "to bind" and in this particular case this verb was the used for enriching classes that exhibit *"binding"*. However, we should further explore if these cases can be generalised for creating axioms using the verb as a relation.

- The analysis carried out in the Gene Ontology revealed that *lexical regularities* do not always play the role of genus. For example, in SNOMED CT the *lexical regularity "congenital stenosis"* plays the role of a target class of an object property that links the class that exhibit the *lexical regularity* and the class that matches with the regularity. Moreover, as it is shown in Figure 6.2, in SNOMED CT the *lexical regularity* does not follow in all the cases the same axiomatic pattern; while `'congenital stenosis of aorta'` is related to the class `'Congenital stenosis (morphologic abnormality)'`, the class `'congenital stenosis of aortic arch'` is related to `'Stenosis (morphologic abnormality)'`.



▲ Abstract
Congenital stenosis
Congenital stenosis of aorta
Congenital stenosis of aortic arch
Congenital stenosis of aortic valve
Congenital stenosis of appendix
Congenital stenosis of aqueduct of Sylv
Congenital stenosis of cardiac valve
Congenital stenosis of carotid artery
Congenital stenosis of cervical canal

Description: 'Congenital stenosis of aorta (disorder)'

Equivalent To

```
'Disease (disorder)'
 and ('Role group (attribute)' some
  (('Associated morphology (attribute)' some 'Congenital stenosis (morphologic
abnormality)')
   and ('Occurrence (attribute)' some 'Congenital (qualifier value)')
   and ('Finding site (attribute)' some 'Aortic structure (body structure)')))
```

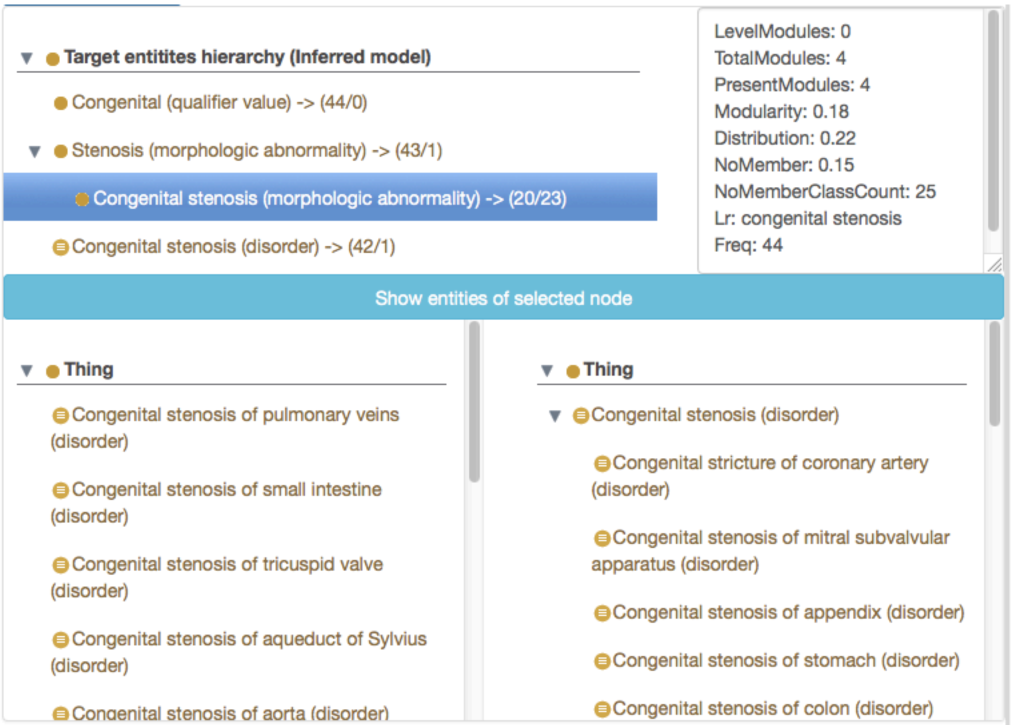
Description: 'Congenital stenosis of aortic arch (disorder)'

Equivalent To

```
'Disease (disorder)'
 and ('Role group (attribute)' some
  (('Associated morphology (attribute)' some 'Stenosis (morphologic abnormality)')
   and ('Occurrence (attribute)' some 'Congenital (qualifier value)')
   and ('Finding site (attribute)' some 'Aortic arch structure (body structure)')))
```

Figure 6.1: Axiomatic description of two terms that exhibit the lexical regularity "congenital stenosis".

Recently, we have combined the modularity metric with the CPE metric so that this helps to measure these kinds of situations. The CPE is provided with a new condition that finds decomposition in the text of a *lexical regularity*, instead of being focused on those classes that exhibit the regularity. We use this decomposition as the input of the modularity metrics, using the decomposed classes as *classes of interest*. Moreover, we expand the modularity metric to use, together with the semantic similarity function, object properties and the inferred model. This can be used to find classes that exhibit a *lexical regularity* and contain axiomatic deviations as it is shown in Figure 6.2. The top left panel of this figure shows the *classes of interest* provided by a new CPE-metric-c4. Together with each *class of interest* there are two numbers: classes exhibiting LR and logically connected to it (left), or not connected (right). For example, 20 classes exhibit *"congenital stenosis"* and are linked with `'congenital stenosis (morphologic abnormality)'`, while 22 are not. In the bottom panels users might further explore these 42 classes: connected on the right and not connected on the left. On the top right panel, quantitative values concerning to the modularity metric are shown. The visualisation of the classes is similar to Protégé ontology editor[1], and all the classes that exhibit the LR are shown according to the original inferred hierarchy. The whole set of *lexical regularities* sorted by different metrics could be explored too, being up to the user to navigate and explore them in detail.



▼ ● **Target entitites hierarchy (Inferred model)**

● Congenital (qualifier value) -> (44/0)

▼ ● Stenosis (morphologic abnormality) -> (43/1)

● Congenital stenosis (morphologic abnormality) -> (20/23)

⊜ Congenital stenosis (disorder) -> (42/1)

LevelModules: 0
TotalModules: 4
PresentModules: 4
Modularity: 0.18
Distribution: 0.22
NoMember: 0.15
NoMemberClassCount: 25
Lr: congenital stenosis
Freq: 44

Show entities of selected node

▼ ● **Thing**

⊜ Congenital stenosis of pulmonary veins (disorder)

⊜ Congenital stenosis of small intestine (disorder)

⊜ Congenital stenosis of tricuspid valve (disorder)

⊜ Congenital stenosis of aqueduct of Sylvius (disorder)

⊜ Congenital stenosis of aorta (disorder)

▼ ● **Thing**

▼ ⊜ Congenital stenosis (disorder)

⊜ Congenital stricture of coronary artery (disorder)

⊜ Congenital stenosis of mitral subvalvular apparatus (disorder)

⊜ Congenital stenosis of appendix (disorder)

⊜ Congenital stenosis of stomach (disorder)

⊜ Congenital stenosis of colon (disorder)

Figure 6.2: Visualisation of deviations observed in the regularity "congenital stenosis".

So far, we have applied the method to a module of SNOMED CT and we pointed out 585 *lexical regularities* that capture deviations between the *lexical regularity* exhibited and axiomatic description. We are currently validating them in collaboration with domain experts in terms of precision and recall. If successfull, we would

[1] `http://protege.stanford.edu/`

apply this new metric to other BioPortal ontologies.

We think that this new metric would capture redundant and missing relations in GO like the ones identified in [Mougin, 2015]. It could be also compared to [Agrawal and Elhanan, 2014], where they propose a lexical method that group SNOMED CT classes in clusters based on common tokens. They compare the relations that concepts grouped in the same cluster have, and use it for proposing missing relations. The study of correspondences between our *lexical regularities* and their clusters could be carried out, as well as studying whether their method for identifying missing restrictions could be used to automatise the creation of ODPs and their codification in OPPL, being the missing piece in our method for automatically contributing to the enrichment of biomedical knowledge resources.

- We think that our method could be coupled to methods that evaluate the quality of an ontology. The hypothesis is that a more axiomatic ontology should have more quality. [Duque-Ramos et al., 2011] present a framework for evaluating the quality of ontologies based on a software quality evaluation standard. We have had the opportunity of adapting the OQuaRE framework in the design of a pipeline that evaluates the evolution of ontologies according to their changes in quality scores [Quesada-Martínez et al., 2015a]. If we formalise the output of our method as a new version of the initial ontology the mentioned pipeline could be applied for evaluating the changes in quality between the initial and the enriched version. Unfortunately, this has not been addressed yet, so it is proposed as a piece of future work.

- Finally, the creation of a repository of ODPs based on lexical content could help users to build ontologies using templates, as it has recently been proposed by the Gene Ontology team with Term Genie [Dietze et al., 2014].

6.4. Publications and contribution in Conferences

6.4.1. Publications JCR

- Quesada-Martínez M., Fernández-Breis J. T., Stevens R. Lexical Characterisation of Bio-Ontologies by the Inspection of Regularities in Labels. Current Bioinformatics. 2015 May 18;10(2):165–76. Available from: http://www.eurekaselect.com/openurl/content.php?genre=article&issn=1574-8936&volume=10&issue=2&spage=165 (Impact Factor 2014: 0,921)

- Quesada-Martínez M., Fernández-Breis J. T., Stevens R., Mikroyannidi E. Prioritising Lexical Patterns to Increase Axiomatisation in Biomedical Ontologies. The role of Localisation and Modularity. Methods of Information in Medicine. 2014;53:1–9. Available from: http://dx.doi.org/10.3414/ME13-02-0026 (Impact Factor 2014: 2,248)

- Quesada-Martínez M., Mikroyannidi E., Fernández-Breis J. T., Stevens R. Approaching the axiomatic enrichment of the Gene Ontology from a lexical perspective. Artificial Intelligence in Medicine. Elsevier B.V.; 2015;(September):1–14. Available from: http://linkinghub.elsevier.com/retrieve/pii/S0933365714001237 (Impact Factor 2014: 2,019)

6.4.2. International Conferences and workshops

- Quesada-Martínez M., Fernandez-Breis J. T. Enrichment of OWL Ontologies : a method for defining axioms from labels. Proceedings of the First International Workshop on Capturing and Re ning Knowledge in the Medical Domain (K-MED 2012). Galway; 2012. p. 1–10.

- Quesada-Martínez M., Fernández-Breis J.T., Stevens R. Extraction and analysis of the structure of labels in biomedical ontologies. Proceedings of the 2nd international workshop on Managing interoperability and compleXity in health systems - MIXHS '12. Maui, Hawaii, US: ACM Press; 2012. p. 7. Available from: http://dl.acm.org/citation.cfm?doid=2389672.2389675

- Quesada-Martínez M., Fernández-Breis J. T., Stevens R. Analysis and Classification of Bio-ontologies by the Structure of their Labels. In: Rojas I, Guzman FMO, editors. International Work-Conference on Bioinformatics and Biomedical Engineering, IWBBIO 2013, Granada, Spain, March 18-20, 2013 Proceedings. Copicentro Editorial; 2013. p. 713–20. Available from: http://iwbbio.ugr.es/papers/iwbbio_113.pdf

- Quesada-Martínez M., Fernández-Breis J. T., Stevens R. Lexical Characterization and Analysis of the BioPortal Ontologies. In: Peek N, Morales RM, Peleg M, editors. Artificial Intelligence in Medicine. Springer Berlin Heidelberg; 2013. p. 206–15. Available from: `http://link.springer.com/chapter/10.1007/978-3-642-38326-7_31`

- Quesada-Martínez M., Fernández-Breis J. T., Stevens R. Lexical Analysis and Characterization of the OBOFoundry Ontologies. The 16th Annual Bio-Ontologies Meeting, co-located with ISMB/ECCB 2013. Berlin; 2013. p. 9–12.

- Mikroyannidi E., Quesada-Martínez M., Tsarkov D., Fernández Breis J. T., Stevens R., Palmisano I. A Quality Assurance Workflow for Ontologies Based on Semantic Regularities. In: Janowicz K, Schlobach S, Lambrix P, Hyvönen E, editors. Knowledge Engineering and Knowledge Management. Springer International Publishing; 2014. p. 288–303. Available from: `http://link.springer.com/10.1007/978-3-319-13704-9_23`

- Quesada-Martínez M., Duque-Ramos A., Fernández-Breis J. T. Analysis of the evolution of ontologies using OQuaRE : Application to EDAM. Proceedings of the International Conference on Biomedical Ontology 2015. Lisbon; 2015. p. 62–6.

Chapter 7

Resumen

La Web Semántica [Tim, Lee et al., 2001, Shadbolt et al., 2006] es la extensión de la *World Wide Web* que permite a las personas compartir contenido más allá de los límites de las aplicaciones y las páginas web[1]. Las tecnologías de la Web Semántica permiten a la gente crear almacenes de datos disponibles online, crear vocabularios, y escribir reglas para la manipulación de datos. Las tecnologías de la web semántica han sido aplicadas en el modelado de las ciencias de la vida dando lugar a los que algunos han denominado usando el término anglosajón *Life Sciences Semantic Web* [Good and Wilkinson, 2006].

Las ontologías son consideradas uno de los pilares básicos de la Web Semántica[2]. En informática, una ontología se define como una especificación formal de una conceptualización compartida. Una ontología es un conjunto de axiomas lógicos diseñados con el objetivo de explicar y justificar el significado de un vocabulario [Borst, 1997]. En otras palabras, una ontología es una representación formal que define categorías de objetos de un dominio de interés y las condiciones que dichos objetos tienen que cumplir para pertenecer a cada una de dichas categorías. Hasta el momento, la definición de Guarino [Guarino, 1998] sobre ontología como artefacto software es una de las más aceptadas [Fernández-Breis, 2003]:

> *"En el sentido filosófico, podemos referirnos a una ontología como un sistema 'particular de categorías que representa una cierta visión del mundo. Como tal, este sistema no depende de un lenguaje particular: la ontología de*

[1] http://semanticweb.org/

[2] http://semanticweb.org/wiki/Ontology.html

> *Aristóteles es siempre la misma, independientemente del lenguaje usado para describirla. Por otro lado, en su uso más típico en IA, una ontología es un artefacto ingenieril constituido por un vocabulario específico para describir una cierta realidad, más un conjunto de supuestos explícitos concernientes al significado pretendido de las palabras del vocabulario. Este conjunto de supuestos tiene generalmente la forma de teorías lógicas de primer orden, donde las palabras del vocabulario aparecen como predicados unarios o binarios, respectivamente llamados conceptos y relaciones. En el caso más simple, una ontología describe una jerarquía de conceptos relacionados por relaciones de subsunción; en los casos más sofisticados, se añaden axiomas para expresar otras relaciones entre conceptos y restringir la posible interpretación."*

En esta definición destaca la codificación de las ontologías como un lenguaje formal (Lenguaje de Representación de Conocimiento) [Stephan et al., 2007]. En concreto, Guarino propone el uso Lenguajes de Representación de Conocimiento basados en teorías lógicas de primer orden. Esta formalización permitirá a los ordenadores procesar el contenido modelado con ontologías y ejecutar algoritmos de razonamiento e inferencia. En este trabajo analizamos diferentes Lenguajes de Representación de Conocimiento y nos centramos en OWL. OWL define varios perfiles con diferentes niveles de expresividad lógica. Cuanto más expresivo es un lenguaje, más complejo será computacionalmente. Las lógicas de primer orden no son decidibles; estos problemas son solucionados usando un sub-conjunto de las lógicas de primer orden llamado Lógica Descriptiva [Baader, 2003]. El perfil OWL DL está basada en esta lógica. Su semántica formal y la disponibilidad de herramientas de razonamiento eficientes han hecho de OWL DL uno de los lenguajes más usados para representar ontologías biomédicas; el lenguaje OBO también pero en sus últimas versiones es posible convertir de OBO a OWL sin perder expresividad.

En los últimos 15 años, la comunidad biomédica ha incrementado sus esfuerzos en el desarrollo de ontologías en ciencias de la vida y no hay ningún motivo para esperar que estos cambien en un futuro [Hoehndorf et al., 2014]. Como consecuencia de su éxito, las ontologías biomédicas son construidas en comunidad con un elevado nivel de actividad. El desarrollo de una ontologías puede ser por tanto el resultado de un trabajo colaborativo entre diferente expertos [Malone and Stevens, 2013]. Dos ejemplos representativos de ontologías en el ámbito de la bioinformática y de la informática médica son Gene Ontology (GO) [Ashburner et al., 2000] y SNOMED CT respectivamente [Cornet and de Keizer, 2008].

Una ontología está formada por cuatro tipos de componentes: conceptos, instancias, relaciones y axiomas. Un concepto definido en el dominio de Gene Ontology es `‘binding’`, que representa la contextualización de la ‘interacción selectiva, no covalente y con frequencia etequiométrica de una molécula con uno o más sitios específicos en otra molécula”. Otro concepto puede ser un tipo más específico de binding, por ejemplo `‘vitamin binding’`. Además, esta relación jerárquica se establece por medio de una relación entre ambos conceptos. Los axiomas son la base lógica de OWL y todo se representa por medio de axiomas.

El desarrollo por parte de expertos en el dominio de ontologías puede dar lugar a artefactos ricos en información del dominio pero con poca axiomatización. En OWL cualquier entidad debe ser referida por un identificador único denominado IRI. Una IRI sería `http://purl.obolibrary.org/obo/GO_0005488`. La última parte de la IRI se conoce también como identificador (GO_0005488). Los identificadores pueden no tener ningún significado como en este caso. Por ejemplo, el identificador anterior está asociado con el concepto `‘binding’` y en ontologías como GALEN el identificador de la IRI sí tiene significado: `http://www.co-ode.org/ontologies/galen#Binding`. OWL también permite separar las IRIs de los identificadores por medio de anotaciones. En este trabajo nos centramos en las etiquetas, cuyo objetivo es proporcionar sin ambigüedad un nombre a un concepto. Existen recomendaciones a la hora de asignar identificador a los conceptos en una ontología [Schober et al., 2009]. Por ejemplo, utilizar un nombrado sistemático de manera que los conceptos que son especializaciones contengan en su identificador el identificador del padre como es el caso de `‘binding’` y su hijo `‘vitamin binding’`. [Third, 2012] estudió los identificadores de las ontologías sobre un corpus de 548 ontologías y detectó, por ejemplo, que este tipo de patrones eran de los más utilizados. Third diferenció entre identificadores simples como `‘binding’` e identificadores complejos como `‘vitamin binding’`, que está formado a su vez por dos identificadores simples. Su estudio reveló que el patrón más utilizado era definir una relación de especialización como la anterior del tipo `subClassOf(AB B)` con 1430 ocurrencias; a esta le siguió en segundo lugar `subClassOf (ABC BC)` entre otras. Sin embargo, puede haber situaciones donde el patrón se siga en el identificador, pero no en su representación axiomática. Por ejemplo, Figure 2.8 muestra que una clase etiquetada como `‘CX3C chemokine receptor binding’` no es descendiente de la clase `‘chemokine receptor binding’` y a este tipo de situaciones es lo que llamamos semántica oculta. La semántica oculta se puede extender más allá de relaciones taxonómicas y podría estar codificando otro tipo de relaciones de asociación.

Trabajos como [Mungall, 2004, Fernandez-Breis et al., 2010, Mungall et al., 2011] han enriquecido Gene Ontology a partir del análisis de sus etiquetas. Sin embargo, estos procesos se centraron en una única ontología y requirieron la intervención manual de los usuarios para crear los nuevos axiomas.

La hipótesis y pregunta de investigación iniciales de esta tesis son:

> *La riqueza axiomática de ontologías biomédicas podría ser mejorada creando nuevos axiomas lógicos, usando información que ya está codificada como parte de las definiciones en lenguaje natural de los conceptos de la ontología.*
>
> *¿Podríamos desarrollar un método automático que transforme semántica oculta en axiomas lógicos, y que sea este método sistemáticamente aplicable para enriquecer repositorios de ontologías biomédicas?*

7.1. Objetivos

El objetivo general de esta tesis es contribuir al análisis de repositorios de conocimiento biomédico ayudando a expertos del dominio a detectar semántica oculta mediante el uso de un método automático que se pueda aplicar de forma sistemática en las ontologías disponibles en repositorios de conocimiento biomédico. Esta metodología ayudará a tener ontologías más completas que exploten la expresividad de los lenguajes formales en los que están definidas. Los objetivos de esta tesis con un mayor nivel de granularidad son:

- Obj1. Desarrollo e implementación de una metodología para la caracterización automática de ontologías usando el análisis de los identificadores descritos en lenguaje natural asociados a sus conceptos.

- Obj2. Desarrollo e implementación de una metodología que permita descubrir semántica oculta y que sea transformable en axiomas lógicos contribuyendo al enriquecimiento de las ontologías biomédicas.

- Obj3. Desarrollo e implementación de una plataforma integrada que ayude a los expertos en el domino con pocos conocimientos técnicos o ontológicos.

7.2. Metodología

La metodología que hemos seguido para el desarrollo de esta tesis doctoral se compone de los siguiente pasos:

- Estudio del estado del arte: web semántica, bioinformática, repositorios de conocimiento biomédico, enriquecimiento de ontologías.
- Formalización de los métodos propuestos en esta tesis.
- Implementación de la metodología y su aplicación con ontologías biomédicas disponibles en internet.
- Análisis de los resultados obtenidos y validación. Debido a la ausencia de un *gold standard* con el cual comparar, se desarrollarán estrategias que establezcan una correspondencia entre nuestro método y otros métodos analizados en el estado del arte. El análisis manual de una comparación automática nos permitirá descubrir fortalezas y debilidades de nuestro método.

7.3. Resultados y conclusiones

La metodología para el análisis sistemático de ontologías a partir del estudio de sus etiquetas va a ser el principal resultado de esta tesis (see Figure 4.1). La metodología propuesta está formada de los siguientes pasos:

- Procesamiento de la ontología y obtención de las regularidades léxicas.
- Módulo de procesamiento de lenguaje natural.
- Módulo basado en métricas.
  - Métricas de modularidad y localidad.
  - Métrica *cross-product extension.*

- Filtrado de regularidades léxicas basado en los valores de las métricas.
- Creación de patrones de enriquecimiento.

A continuación presentamos los resultados haciendo referencia a los trabajos presentados como compendio de esta tesis doctoral:

- *Lexical characterization of Bio-Ontologies by the inspection of regularities in labels*

  Se realiza un estudio de la estructura de las etiquetas en ontologías disponibles en BioPortal, haciendo uso de las métricas definidas en la metodología y que se clasifican en términos del interés potencial para su enriquecimiento axiomático.

  El ratio identificador/clases está cerca de 1, lo que significa que los creadores de las ontologías sí añaden identificadores que describen en lenguaje natural los conceptos representados en el dominio. 65 % de las ontologías usan etiquetas como identificadores. Esto supone un cambio ya que en trabajos previos la opción más usada era incluirlos como fragmentos de las IRI.

  Las ontologías biomédicas son ricas en regularidades léxicas. Usando un porcentaje de cobertura del 1 % se obtienen 8175 regularidades en 118 ontologías biomédicas. Esto es un indicador de que las ontologías biomédicas utilizan un nombrado sistemático. El procesamiento con éxito de las 118 ontologías demuestra la escalabilidad del método.

  75 % de los axiomas en las ontologías de BioPortal son anotaciones o `rdfs:subClassOf`. Esto es un indicador de que las ontologías biomédicas son en gran medida vocabularios controlados y taxonomías planas por la que la aplicación de métodos de enriquecimientos contribuiría a incrementar su expresividad.

  77.22 % de ontologías en BioPortal no reutilizan conceptos de otras ontologías. Sin embargo, 15.60 % de las regularices léxicas se corresponden con etiquetas completas de otras ontologías en BioPortal, lo que indica que las clases representadas por este porcentaje podrían ser descompuestas y relacionadas con otras a través de relaciones.

  Nuestro método define una ontología adecuada para su enriquecimiento como aquella que sus etiquetas contienen regularidades, dichas regularidades son generales y afectan a un porcentaje elevado de clases, y además esas regularidades tienen correspondencias en otras ontologías externas. Usando esta información, se aplica un algoritmo de clustering usando como datos de entrada la caracterización léxica de un conjunto de ontologías de BioPortal; estas ontologías son son clasificadas en tres clusters según su adecuación a ser usadas en procesos de enriquecimiento. De acuerdo con la clasificación de estos clusters, el 75 % de las ontologías de BioPortal analizadas se podrían beneficiar de procesos de enriquecimiento basados en las regularidades léxicas.

Por último, 23.49 % de las regularidades aparecen en más de una ontología. Este valor sugiere que la creación de axiomas basados en estas relaciones podría ser reutilizada para sistemáticamente repetir el proceso entre ontologías.

- *Prioritizing lexical patterns to increase axiomatisation in biomedical ontologies*:

  En este trabajo proponemos métricas para sugerir qué regularidades léxicas deberían ser el punto de partida para definir los patrones de enriquecimiento. Esta priorización se modela usando métricas. Una métrica es una función que recibe una regularidad léxica como parámetro y genera un valor entre m y n. En este trabajo se definen dos métricas:

  (1) Métricas basadas en la localización de las clases que exhiben una regularidad léxica. Para ello se utilizan funciones que miden la distancia semántica entre clases. Estas funciones tienen en cuenta las relaciones jerárquicas.

  (2) Usando distancia semántica se mide la distribución de las regularidades léxicas respecto a un conjunto de clases especificado como parámetro de entrada.

  Estas métricas proporcionan información sobre los principios de ingeniería seguidos en el desarrollo de las ontologías y permiten ordenar las regularidades. El método y estas métricas permiten sugerir relaciones entre clases que no han sido explícitamente codificadas en la ontología.

- *Approaching the axiomatic enrichment of the Gene Ontology from a lexical perspective*

  El objetivo principal de este trabajo es ser capaces de medir cómo las regularidades léxicas pueden ser usadas para la creación automática de relaciones formales entre clases y evaluar los resultados aplicando nuestro método en Gene Ontology. La elección de Gene Ontology como caso de uso también se debe a su extendido uso entre la comunidad biomédica (las clases de Gene Ontology han sido usadas para producir millones de anotaciones que están disponibles en bases de datos de anotaciones como [Barrell et al., 2009]. Además, Gene Ontology es rica en regularidades y alineamientos en ontologías externas. De hecho, fue seleccionada como miembro de aquellos clusters con ontologías prometedoras para su enriquecimiento en el primer trabajo de caracterización léxica de BioPortal .

En este trabajo se incluye en nuestro método una nueva métrica: *cross-product extension* (CPE). Esta métrica pretende estimar el potencial interés de una regularidad usando información de matches externos presentes en las clases que exhiben la regularidad. Por ejemplo, la regularidad *"binding"* es exhibida por clases como `'vitamin binding'`. Como hemos explicado, `'vitamin binding'` es un identificador complejo. Tanto `'binding'` como `'vitamin binding'` son clases en GO, y además siguen un naming respaladado por la relación `rdfs:subClassOf` que hay entre ellas. Sin embargo, ¿qué hace diferente al `'vitamin binding'` de otros descendientes de `'binding'` como `'alcohol binding'`? Un alineamiento parcial de las etiquetas de las clases que exhiben una regularidad puede ayudarnos a resolver este problema. Por ejemplo, tanto `'vitamin'` como `'alcohol'` son clases en la ontología *Chemical Entities of Biological Interest* (ChEBI), y si aplicamos la métrica CPE obtenemos que un 20 % de las clases que exhiben la regularidad *"binding"* son totalmente descomponibles en clases de ChEBI más el `'binding'`, por lo que un enriquecimiento axiomático de dichas clases sería adecuado como se hizo en [Fernandez-Breis et al., 2010] con el patrón "X binding" (Figura 1.3).

El consorcio de Gene Ontology recientemente ha enriquecido GO usando los llamados productos cruzados. Los productos cruzados usan como base GO y otra ontología externa para crear clases definidas combinado clases de ambas. Nosotros aplicamos nuestro método y la CPE a Gene Ontology y estudiamos cómo nuestro análisis léxico identifica y reconstruye los productos cruzados definidos por el consorcio de Gene Ontology. Para ello formalizamos la relación entre las regularidades léxicas y los productos cruzados. Los productos cruzados se basan en la definición aristotélica *genus-differentia* del tipo *"un X es un G que D"*. Aquí, X es la clases a definir, G es el *genus* (la clase más general) y D es la *differentia*. En nuestro ejemplo, `‘binding’` sería el *genus* y `‘vitamin’` o `‘alcohol’` las diferencias. Aunque nuestro método y el desarrollado por el consorcio de Gene Ontology no son exactamente iguales, ambos tienen un objetivo similar. Nosotros queremos contrastar la hipótesis de que las regularidades léxicas pueden ser usadas para capturar *genus* de forma automática, por ello formalizamos la relación entre los dos y los comparamos en términos de exhaustividad y precisión.

Como media, la métrica CPE reveló que un 31.48 % de las clases que exhiben regularidades léxicas contienen fragmentos que son clases en dos ontologías externas *Cell Ontology* y CheBI, además un 18.90 % de estas clases pueden ser totalmente descompuestas en fragmentos más pequeños que son clases (identificadores complejos).

Nuestros resultados muestran que la métrica CPE permite a nuestro método detectar productos cruzados con una exhaustividad y precisión media del 62 % y del 28 % respectivamente. El estudio es completado con un análisis de los falsos positivos para encontrar una explicación al bajo valor de precisión obtenido.

A continuación se enumeran algunas de las contribuciones de esta tesis:

- La metodologías para analizar ontologías a partir de las regularidades léxicas en los identificadores.
- Un método escalable debido a:
  - La organización de los identificadores como un grafo de tokens. Este grafo permite acelerar el proceso de búsqueda de las regularidades léxicas y utiliza parámetros como el porcentaje de cobertura como mecanismo para optimizar y podar las búsquedas.
  - Las métricas que permiten la prioritización de regularidades léxicas usando como base diferentes aspectos relacionados con propiedades semánticas de las ontologías. Por ejemplo, la distancia semántica modularidad o alineamientos que usan técnicas de semejanza léxica.
  - El grafo nos permite implementar un algoritmos de alineamiento entre ontologías basado en alineamientos parciales en lugar de la etiqueta completa y usando técnicas de pre-procesamiento de lenguaje natural.

- La aplicación del método sobre un conjunto de ontologías biomédicas disponibles en BioPortal para:
  - Caracterizarlas léxicamente usando el contenido en lenguaje natural de sus identificadores y matches entre las regularidades léxicas y otras ontologías. El método permite crear clusters de ontologías según su adecuación para ser usadas para su enriquecimiento.
  - Aplicación del método sobre Gene Ontology y reconstrucción de los productos cruzados previamente usados por el GO Consortium para enriquecerlo con el objetivo de evaluar la metodología.

En resumen, el método contribuye al análisis automático y sistemático de ontologías biomédicas. El método permite el análisis de las regularidades léxicas desde diferentes ejes que son seleccionados mediante la aplicación de métricas. Sin embargo, los experimentos usando estas métricas han revelado que la semántica oculta detrás de una regularidad léxica puede ser de diferentes tipos. Mientras sub/super-regularidades que son clases pueden ser usadas para crear relaciones jerárquicas, y esto puede ser cuantificado con la métrica de localidad, regularidades léxicas que son verbos pueden ser usadas para crear otros tipos de relaciones. Estas diferencias deben ser consideradas a la hora de automatizar la transformación de regularidades léxicas en patrones de diseño ontológicos, por lo que su mejora se propone como parte del trabajo futuro. Otras líneas de trabajo futuro son evaluar como el enriquecimiento axiomático afecta a la calidad de la ontología, así como la creación de un repositorio de patrones de conocimiento ontológico reutilizables basados en las regularidad léxicas.

La inspección de las regularidades léxicas ayuda a expertos en el dominio con pocos conocimientos semánticos en la creación de axiomas lógicos. Todo los métodos propuestos en esta tesis han sido implementados y están disponibles en la aplicación web `http://sele.inf.um.es/ontoenrich`. La herramienta permite realizar el análisis léxico de una ontología, navegar por sus regularidades léxicas y aplicar métricas explicadas para la prioritización y el análisis avanzado de las regularidades.

Bibliography

[Abney, 2007] Abney, S. (2007). *Semisupervised Learning for Computational Linguistics.* Chapman & Hall CRC.

[Agrawal and Elhanan, 2014] Agrawal, A. and Elhanan, G. (2014). Contrasting lexical similarity and formal definitions in SNOMED CT: Consistency and implications. *Journal of Biomedical Informatics*, 47:192–198.

[Ashburner et al., 2000] Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G. (2000). Gene Ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25–29.

[Auger and Barrière, 2008a] Auger, A. and Barrière, C., editors (2008a). *Pattern-based Approaches to Semantic Relation Extraction*, volume 14. John Benjamins Publishing Company.

[Auger and Barrière, 2008b] Auger, A. and Barrière, C. (2008b). Pattern-based approaches to semantic relation extraction: A state-of-the-art. *Terminology*, 14(1):1–19.

[Aussenac-Gilles and Jacques, 2008] Aussenac-Gilles, N. and Jacques, M.-P. (2008). Designing and evaluating patterns for relation acquisition from texts with Caméléon. *Terminology*, 14(1):45–73.

[Baader, 2003] Baader, F. (2003). *The description logic handbook: theory, implementation, and applications.*

[Baader et al., 2008] Baader, F., Horrocks, I., and Sattler, U. (2008). Chapter 3 Description Logics. In *Handbook of Knowledge Representation*, volume 3, pages 135–179.

[Bada and Hunter, 2007] Bada, M. and Hunter, L. (2007). Enrichment of OBO Ontologies. *J. of Biomedical Informatics*, 40(3):300–315.

[Bard et al., 2005] Bard, J., Rhee, S. Y., and Ashburner, M. (2005). An ontology for cell types. *Genome biology*, 6(2):R21.

[Barrell et al., 2009] Barrell, D., Dimmer, E., Huntley, R. P., Binns, D., ODonovan, C., and Apweiler, R. (2009). The GOA database in 2009 - An integrated Gene Ontology Annotation resource. *Nucleic Acids Research*, 37(SUPPL. 1):396–403.

[Benson et al., 2007] Benson, D. a., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., and Wheeler, D. L. (2007). GenBank. *Nucleic Acids Research*, 35(Database):D21–D25.

[Berman et al., 2000] Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. (2000). The Protein Data Bank. *Nucleic acids research*, 28(1):235–242.

[Bodenreider, 2004] Bodenreider, O. (2004). The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic acids research*, 32(Database issue):D267–D270.

[Borst, 1997] Borst, W. N. (1997). *Construction of engineering ontologies for knowledge sharing and reuse.* PhD thesis, University of Twente.

[Bravo et al., 2014] Bravo, A., Pinero, J., Queralt, N., Rautschka, M., and Furlong, L. I. (2014). Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research. Technical report.

[Brewster, 2006] Brewster, C. (2006). Ontology Learning from Text: Methods, Evaluation and Applications Paul Buitelaar, Philipp Cimiano, and Bernado Magnini (Editors). *Computational Linguistics*, 32(4):569–572.

[Bult et al., 2007] Bult, C. J., Eppig, J. T., Kadin, J. A., Richardson, J. E., and Blake, J. A. (2007). The Mouse Genome Database (MGD): Mouse Biology and Model Systems. *Nucleic Acids Research*, 36(Database):D724–D728.

[Campbell et al., 1998] Campbell, K. E., Tuttle, M. S., and Spackman, K. a. (1998). A "lexically-suggested logical closure" metric for medical terminology maturity. *Proceedings of the AMIA Annual Symposium*, pages 785–789.

[Ceusters, 2006] Ceusters, W. (2006). Towards A Realism-Based Metric for Quality Assurance in Ontology Matching. In Bennett, B. and Fellbaum, C., editors, *Formal Ontology in Information Systems, Proceedings of the Fourth International Conference, FOIS 2006, November 9-11*, volume 150 of *Frontiers in Artificial Intelligence and Applications*, pages 321–332, Baltimore, Maryland, USA,. IOS Press.

[Ceusters et al., 2004] Ceusters, W., Smith, B., Kumar, A., and Dhaen, C. (2004). Ontology-based error detection in SNOMED-CT. *Proceedings of MEDINFO*, 2004:482–486.

[Cimiano et al., 2009] Cimiano, P., Mädche, A., Staab, S., and Völker, J. (2009). Ontology Learning. In *Handbook on Ontologies*, pages 245–267. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Consortium, 2001] Consortium, T. G. O. (2001). Creating the Gene Ontology Resource: Design and Implementation. *Genome Research*, 11(8):1425–1433.

[Consortium, 2012] Consortium, T. U. (2012). Reorganizing the Protein Space at the Universal Protein Resource (UniProt). *Nucleic Acids Research*, 40(D1):D71–D75.

[Consortium, 2015] Consortium, T. U. (2015). UniProt: a hub for Protein information. *Nucleic Acids Research*, 43(D1):D204–D212.

[Cornet and de Keizer, 2008] Cornet, R. and de Keizer, N. (2008). Forty years of SNOMED: a literature review. *BMC Medical Informatics and Decision Making*, 8(Suppl 1):S2.

[D'Aquin and Motta, 2011] D'Aquin, M. and Motta, E. (2011). Watson, More Than a Semantic Web Search Engine. *Semantic Web*, 2:55–63.

[De Coronado et al., 2009] De Coronado, S., Wright, L. W., Fragoso, G., Haber, M. W., Hahn-Dantona, E. A., Hartel, F. W., Quan, S. L., Safran, T., Thomas, N., and Whiteman, L. (2009). The NCI Thesaurus Quality Assurance life cycle. *Journal of biomedical informatics*, 42(3):530–9.

[Degtyarenko et al., 2007] Degtyarenko, K., de Matos, P., Ennis, M., Hastings, J., Zbinden, M., McNaught, a., Alcantara, R., Darsow, M., Guedj, M., and Ashburner, M. (2007). ChEBI: a database and ontology for chemical entities of biological interest. *Nucleic Acids Research*, 36(Database):D344–D350.

[Dentler et al., 2011] Dentler, K., Cornet, R., ten Teije, A., and de Keizer, N. (2011). Comparison of Reasoners for Large Ontologies in the OWL 2 EL Profile. *Semant. web*, 2(2):71–87.

[Dietze et al., 2014] Dietze, H., Berardini, T. Z., Foulger, R. E., Hill, D. P., Lomax, J., OsumiSutherland, D., Roncaglia, P., and Mungall, C. J. (2014). TermGenie - A web application for pattern-based ontology class generation. *Journal of Biomedical Semantics*, 5(1):48.

[Ding et al., 2004] Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R. S., Peng, Y., Reddivari, P., Doshi, V., and Sachs, J. (2004). Swoogle. In *Proceedings of the Thirteenth ACM conference on Information and knowledge management - CIKM '04*, page 652, New York, New York, USA. ACM Press.

[Duque-Ramos et al., 2011] Duque-Ramos, A., Fernández-Breis, J. T., Stevens, R., and Aussenac-Gilles, N. (2011). OQuaRE: A square-based approach for evaluating the quality of ontologies. *Journal of Research and Practice in Information Technology*, 43(2):159–176.

[Egaña Aranguren et al., 2008] Egaña Aranguren, M., Wroe, C., Goble, C., and Stevens, R. (2008). In situ migration of handcrafted ontologies to reason-able forms. *Data & Knowledge Engineering*, 66(1):147–162.

[Egaña-Aranguren, 2009] Egaña-Aranguren, M. (2009). *Role and Application of Ontology Design Patterns in Bio-Ontologies.* PhD thesis, The University of Manchester.

[Eilbeck et al., 2005] Eilbeck, K., Lewis, S. E., Mungall, C. J., Yandell, M., Stein, L., Durbin, R., and Ashburner, M. (2005). The Sequence Ontology: a tool for the unification of genome annotations. *Genome biology*, 6(5):R44.

[Euzenat and Shvaiko, 2011] Euzenat, J. and Shvaiko, P. (2011). *Ontology matching.* Springer-Verlag New York, Inc.

[Faria et al., 2013a] Faria, D., Pesquita, C., Santos, E., Cruz, I. F., and Couto, F. M. (2013a). AgreementMakerLight Results for OAEI 2013. *ISWC Workshop*.

[Faria et al., 2014] Faria, D., Pesquita, C., Santos, E., Cruz, I. F., and Couto, F. M. (2014). Automatic Background Knowledge Selection for Matching Biomedical Ontologies. {*PLoS*} {*ONE*}, 9(11):e111226.

[Faria et al., 2013b] Faria, D., Pesquita, C., Santos, E., Palmonari, M., Cruz, I. F., and Couto, F. M. (2013b). The AgreementMakerLight ontology matching system. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8185 LNCS:527–541.

[Fauconnier et al., 2015] Fauconnier, J.-P., Kamel, M., and Rothenburger, B. (2015). A supervised machine learning approach for taxonomic relation recognition through non-linear enumerative structures. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing - SAC '15*, pages 423–425, New York, New York, USA. ACM Press.

[Fernández-Breis, 2003] Fernández-Breis, J. T. (2003). *Un Entorno de Integración de Ontologías par el Desarrollo de Sistemas de Gestión de Conocimiento*. PhD thesis, University of Murcia.

[Fernandez-Breis et al., 2010] Fernandez-Breis, J. T., Iannone, L., Palmisano, I., Rector, A. L., and Stevens, R. (2010). Enriching the Gene Ontology via the Dissection of Labels Using the Ontology Pre-processor Language. In Cimiano, P. and Pinto, H. S., editors, *Knowledge Engineering and Management by the Masses*, number 6317 in Lecture Notes in Computer Science, pages 59–73. Springer Berlin Heidelberg.

[Fernandez-Lopez et al., 1999] Fernandez-Lopez, M., Fernandez-Lopez, M., Gomez-Perez, A., Gomez-Perez, A., Sierra, J. P., Sierra, J. P., Sierra, a. P., and Sierra, a. P. (1999). Building a Chemical Ontology Using Methonology and the Ontology Design Environment. *IEEE Intelligent Systems*, 14(1):37–46.

[Friedman et al., 2006] Friedman, C., Borlawsky, T., Shagina, L., Xing, H. R., and Lussier, Y. a. (2006). Bio-ontology and text: Bridging the modeling gap. *Bioinformatics*, 22(19):2421–2429.

[Gangemi, 2005] Gangemi, A. (2005). Ontology Design Patterns for Semantic Web Content. In Gil, Y., Motta, E., Benjamins, V. R., and Musen, M. A., editors, *The Semantic Web – ISWC 2005*, volume 3729 of *Lecture Notes in Computer Science*, pages 262–276.

[Gangemi and Presutti, 2009] Gangemi, A. and Presutti, V. (2009). Ontology Design Patterns. In *Handbook on Ontologies*, pages 221–243. Springer Berlin Heidelberg, Berlin, Heidelberg.

[García et al., 2010] García, J., García-Peñalvo, F. J., and Therón, R. (2010). A Survey on Ontology Metrics. In Lytras, M. D., Ordonez De Pablos, P., Ziderman, A., Roulstone, A., Maurer, H., and Imber, J. B., editors, *Knowledge Management, Information Systems, E-Learning, and Sustainability Research*, volume 111, pages 22–27. Springer Berlin Heidelberg.

[Genesereth and Nilsson, 1987] Genesereth, M. R. and Nilsson, N. J. (1987). *Logical Foundations of Artificial Intelligence.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[Golbreich et al., 2013] Golbreich, C., Grosjean, J., and Darmoni, S. J. (2013). The Foundational Model of Anatomy in OWL 2 and its use. *Artificial Intelligence in Medicine*, 57(2):119–132.

[Golbreich et al., 2007] Golbreich, C., Horridge, M., Horrocks, I., Motik, B., and Shearer, R. (2007). OBO and OWL: Leveraging Semantic Web Technologies for the Life Sciences. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 4825 LNCS, pages 169–182.

[Gómez-Pérez, 1999] Gómez-Pérez, A. (1999). Ontological engineering: a state of the art. *Expert Update: Knowledge Based Systems and Applied Artificial Intelligence*, 2(3):33–43.

[Gonzalez and Dankel, 1993] Gonzalez, A. J. and Dankel, D. D. (1993). *The Engineering of Knowledge-based Systems: Theory and Practice.* Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

[Good and Wilkinson, 2006] Good, B. M. and Wilkinson, M. D. (2006). The Life Sciences Semantic Web is full of creeps! *Briefings in Bioinformatics*, 7(3):275–286.

[Grau et al., 2008] Grau, B. C., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P., and Sattler, U. (2008). OWL 2: The next step for OWL. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(4):309–322.

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

[Gruninger and Fox, 1996] Gruninger, M. and Fox, M. S. (1996). The Logic of Enterprise Modelling. In *Modelling and Methodologies for Enterprise Integration*, pages 140–157. Springer US, Boston, MA.

[Guarino, 1995] Guarino, N. (1995). Formal Ontology, Conceptual Analysis and Knowledge Representation. *Int. J. Hum.-Comput. Stud.*, 43(5-6):625–640.

[Guarino, 1998] Guarino, N. (1998). Formal Ontology and Information Systems. *Fois'98*, 46(June):3–15.

[Hagen, 2000] Hagen, J. B. (2000). The origins of bioinformatics. *Nature reviews. Genetics*, 1(3):231–236.

[Hearst, 1992] Hearst, M. a. (1992). Automatic Acquisition of Hyponyms ftom Large Text Corpora. *Proceedings of the 14th conference on Computational Linguistics*, 2:23–28.

[Hoehndorf et al., 2014] Hoehndorf, R., Haendel, M., Stevens, R., and Rebholz-Schuhmann, D. (2014). Thematic series on biomedical ontologies in JBMS: challenges and new directions. *Journal of biomedical semantics*, 5:15.

[Hoehndorf et al., 2015] Hoehndorf, R., Slater, L., Schofield, P. N., and Gkoutos, G. V. (2015). Aber-OWL: a framework for ontology-based data access in biology. pages 1–9.

[Horrocks et al., 2003] Horrocks, I., Patel-Schneider, P., and van Harmelen, F. (2003). From SHIQ and RDF to OWL:The Making of a Web Ontology Language - Google Search. *Journal of Web Semantics*, 1(1):7–26.

[Hubbard et al., 2002] Hubbard, T., Barker, D., Birney, E., Cameron, G., Chen, Y., Clark, L., Cox, T., Cuff, J., Curwen, V., Down, T., Durbin, R., Eyras, E., Gilbert, J., Hammond, M., Huminiecki, L., Kasprzyk, a., Lehvaslaiho, H., Lijnzaad, P., Melsopp, C., Mongin, E., Pettett, R., Pocock, M., Potter, S., Rust, a., Schmidt, E., Searle, S., Slater, G., Smith, J., Spooner, W., Stabenau, a., Stalker, J., Stupka, E., Ureta-Vidal, a., Vastrik, I., and Clamp, M. (2002). The Ensembl genome database project. *Nucleic acids research*, 30(1):38–41.

[Kamdar et al., 2015] Kamdar, M. R., Tudorache, T., and Musen, M. A. (2015). Investigating Term Reuse and Overlap in Biomedical Ontologies. In *Proceedings of the ICBO 2015 International Conference on Biomedical Ontology 2005*, pages 51–55.

[Kocbek et al., 2012] Kocbek, S., Perret, J.-l., and Kim, J.-d. (2012). Visual presentation of mappings between biomedical ontologies Visualization of BioPortal Mapping Data. *Proceedings of the 5th International Workshop on Semantic Web Applications and Tools for Life Sciences.*

[Legaz-García, 2015] Legaz-García, M. d. C. (2015). *Integración de Información Biomédica Basada en Tecnologías Semánticas Avanzadas.* PhD thesis, Universidad de Murcia.

[Lenat and Guha, 1989] Lenat, D. B. and Guha, R. V. (1989). *Building Large Knowledge-Based Systems; Representation and Inference in the Cyc Project.* Addison-Wesley Longman Publishing Co., Inc.

[Liddy, 2001] Liddy, E. D. (2001). Natural Language Processing. In *Encyclopedia of Library and Information Science, 2nd Ed.* Marcel Decker, Inc., NY.

[Liu et al., 2011a] Liu, K., Chapman, W. W., Savova, G., Chute, C. G., Sioutos, N., and Crowley, R. S. (2011a). Effectiveness of lexico-syntactic pattern matching for ontology enrichment with clinical documents. *Methods of Information in Medicine*, 50(5):397–407.

[Liu et al., 2011b] Liu, K., Hogan, W. R., and Crowley, R. S. (2011b). Natural Language Processing methods and systems for biomedical ontology learning. *Journal of Biomedical Informatics*, 44(1):163–179.

[Liu et al., 2013] Liu, K., Mitchell, K. J., Chapman, W. W., Savova, G. K., Sioutos, N., Rubin, D. L., and Crowley, R. S. (2013). Formative evaluation of ontology learning methods for entity discovery by using existing ontologies as reference standards. *Methods of Information in Medicine*, 52(4):308–16.

[Liu et al., 2012] Liu, Y., Bill, R., Fiszman, M., Rindflesch, T., Pedersen, T., Melton, G. B., and Pakhomov, S. V. (2012). Using SemRep to label semantic relations extracted from clinical text. *AMIA Annual Symposium proceedings*, 2012:587–95.

[Lord, 2010] Lord, P. (2010). Components of an Ontology. Ontogenesis. http://ontogenesis.knowledgeblog.org/514.

[Lozano-Tello and Gómez-Pérez, 2004] Lozano-Tello, A. and Gómez-Pérez, A. (2004). Ontometric: A method to choose the appropriate ontology. *Journal of Database Management*, 2:1–18.

[Luscombe et al., 2001] Luscombe, N. M., Greenbaum, D., and Gerstein, M. (2001). What is bioinformatics? A proposed definition and overview of the field. *Methods of Information in Medicine*, 40(4):346–358.

[Machado et al., 2015] Machado, C. M., Rebholz-Schuhmann, D., Freitas, A. T., and Couto, F. M. (2015). The semantic web in translational medicine: current applications and future directions. *Briefings in Bioinformatics*, 16(1):89–103.

[Malone and Stevens, 2013] Malone, J. and Stevens, R. (2013). Measuring the level of activity in community built bio-ontologies. *Journal of Biomedical Informatics*, 46(1):5–14.

[Manning et al., 2014] Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. Association for Computational Linguistics.

[Martínez-Costa, 2011] Martínez-Costa, C. (2011). *Modelos de representación y transformación para la interoperabilidad semántica entre estándares de Historia Clínica*

Electrónica basados en arquitectura de modelo dual. PhD thesis, Universidad de Murcia.

[Mikroyannidi et al., 2011] Mikroyannidi, E., Iannone, L., Stevens, R., and Rector, A. (2011). Inspecting Regularities in Ontology Design Using Clustering. In Aroyo, Lora and Welty, Chris and Alani, Harith and Taylor, Jamie and Bernstein, Abraham and Kagal, Lalana and Noy, Natasha and Blomqvist, E., editor, *The Semantic Web – ISWC 2011*, volume 7031 of *Lecture Notes in Computer Science*, pages 438–453. Springer Berlin Heidelberg.

[Miles et al., 2005] Miles, A., Matthews, B., Wilson, M., and Brickley, D. (2005). SKOS core: simple knowledge organisation for the web. In *Proceedings of the 2005 International Conference on Dublin Core and Metadata Applications: Vocabularies in Practice*, pages 1:–9, Madrid, Spain. Dublin Core Metadata Initiative.

[Mossakowski et al., 2014] Mossakowski, T., Kutz, O., and Codescu, M. (2014). Ontohub: A semantic repository for heterogeneous ontologies. In *Proc. of the Theory Day in Computer Science (DACS-2014), satellite workshop of ICTAC-2014*, University of Bucharest.

[Mougin, 2015] Mougin, F. (2015). Identifying Redundant and Missing Relations in the Gene Ontology. In *Digital Healthcare Empowering Europeans*, pages 195–199.

[Mungall, 2004] Mungall, C. J. (2004). Obol: Integrating Language and Meaning in Bio-ontologies: Conference Papers. *Comp. Funct. Genomics*, 5(6-7):509–520.

[Mungall et al., 2011] Mungall, C. J., Bada, M., Berardini, T. Z., Deegan, J., Ireland, A., Harris, M. a., Hill, D. P., and Lomax, J. (2011). Cross-product extensions of the Gene Ontology. *Journal of Biomedical Informatics*, 44(1):80–86.

[Musen et al., 2012] Musen, M. a., Noy, N. F., Shah, N. H., Whetzel, P. L., Chute, C. G., Story, M.-a., and Smith, B. (2012). The National Center for Biomedical Ontology. *Journal of the American Medical Informatics Association*, 19:190–195.

[Nasraoui, 2008] Nasraoui, O. (2008). Web data mining. *ACM SIGKDD Explorations Newsletter*, 10(2):23.

[National, 2015] National, U. o. M. (2015). SPECIALIST Lexicon (http://www.nlm.nih.gov/pubs/factsheets/umlslex.html).

[Navigli and Velardi, 2004] Navigli, R. and Velardi, P. (2004). Structural Semantic Interconnection: a Knowledge-Based Approach to Word Sense Disambiguation. *Proceedings of SENSEVAL-3 Workshop (SENSEVAL) in the 42th Annual Meeting of the Association for Computational Linguistics (ACL 2004)*, (July):179–182.

[Neches et al., 1991] Neches, R., Fikes, R., Finin, T., Gruber, T., Patil, R., Senator, T., and Swartout, W. R. (1991). Enabling Technology for Knowledge Sharing. *Ai Magazine*, 12(3):36.

[Nor Azlinayati Abdul et al., 2010] Nor Azlinayati Abdul, M., Sean, B., and Stevens, R. (2010). A Survey of Identifiers and Labels in OWL Ontologies. In *OWLED'10*.

[Pacheco et al., 2009] Pacheco, E., Stenzhorn, H., Nohama, P., Paetzold, J., and Schulz, S. (2009). Detecting Underspecification in SNOMED CT concept definitions through natural language processing. *AMIA. Annual Symposium proceedings / AMIA Symposium. AMIA Symposium*, 2009:492–496.

[Pesquita, 2012] Pesquita, C. (2012). *Automated Extension of Biomedical Ontologies*. PhD thesis, Universidade de Lisboa.

[Pesquita et al., 2009] Pesquita, C., Faria, D., Falcão, A. O., Lord, P., and Couto, F. M. (2009). Semantic Similarity in Biomedical Ontologies. *PLoS Computational Biology*, 5(7):e1000443.

[Poveda-Villalón et al., 2012] Poveda-Villalón, M., Suárez-Figueroa, M. C., and Gómez-Pérez, A. (2012). Validating Ontologies with OOPS! In Ten Teije, A., Völker, J., Handschuh, S., Stuckenschmidt, H., D'Acquin, M., Nikolov, A., and Aussenac-Gilles, Nathalie and Hernandez, N., editors, *Knowledge Engineering and Knowledge Management*, volume 7603, pages 267–281. Springer Berlin Heidelberg.

[Power, 2010] Power, R. (2010). Complexity assumptions in ontology verbalisation. In *Proceedings of the ACL 2010 Conference Short Papers*, number July, pages 132–136.

[Pruitt, 2004] Pruitt, K. D. (2004). NCBI Reference Sequence (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Research*, 33(Database issue):D501–D504.

[Quesada-Martínez et al., 2015a] Quesada-Martínez, M., Duque-Ramos, A., and Fernández-Breis, J. T. (2015a). Analysis of the evolution of ontologies using OQuaRE: Application to EDAM. In *Proceedings of the International Conference on Biomedical Ontology 2015*, pages 62–66, Lisbon.

[Quesada-Martínez et al., 2015b] Quesada-Martínez, M., Fernández-Breis, J., and Stevens, R. (2015b). Lexical Characterisation of Bio-Ontologies by the Inspection of Regularities in Labels. *Current Bioinformatics*, 10(2):165–176.

[Quesada-martínez et al., 2013] Quesada-martínez, M., Fernández-breis, J. T., and Stevens, R. (2013). Lexical Analysis and Characterization of the OBOFoundry Ontologies. In *The 16th Annual Bio-Ontologies Meeting, co-located with ISMB/ECCB 2013*, pages 9–12, Berlin.

[Quesada-Martínez et al., 2015c] Quesada-Martínez, M., Fernández-Breis, J. T., Stevens, R., and Aussenac-Gilles, N. (2015c). OntoEnrich: A Platform for the Lexical Analysis of Ontologies. In Lambrix, P., Hyvönen, E., Blomqvist, E., Presutti, V., Qi, G., Sattler, U., Ding, Y., and Ghidini, C., editors, *Knowledge Engineering and Knowledge Management*, volume 8982 of *Lecture Notes in Computer Science*, pages 172–176. Springer International Publishing.

[Quesada-Martínez et al., 2014] Quesada-Martínez, M., Fernández-Breis, J. T., Stevens, R., and Mikroyannidi, E. (2014). Prioritising Lexical Patterns to Increase Axiomatisation in Biomedical Ontologies. The role of Localisation and Modularity. *Methods of Information in Medicine*, 53:1–9.

[Quesada-Martínez et al., 2015d] Quesada-Martínez, M., Mikroyannidi, E., Fernández-Breis, J. T., and Stevens, R. (2015d). Approaching the axiomatic enrichment of the Gene Ontology from a lexical perspective. *Artificial Intelligence in Medicine*, (September):1–14.

[Rector and Iannone, 2012] Rector, A. and Iannone, L. (2012). Lexically suggest, logically define: Quality assurance of the use of qualifiers and expected results of post-coordination in SNOMED CT. *Journal of Biomedical Informatics*, 45(2):199–209.

[Rector et al., 2011] Rector, A. L., Brandt, S., and Schneider, T. (2011). Getting the foot out of the pelvis: modeling problems affecting use of SNOMED CT hierarchies in practical applications. *Journal of the American Medical Informatics Association: JAMIA*, 18(4):432–440.

[Rodríguez-García, 2014] Rodríguez-García, M. Á. (2014). *Extracción Semántica de Información basada en Evolución de Ontologías*. PhD thesis, Universidad de Murcia.

[Rogers, 2006] Rogers, J. E. (2006). Quality assurance of medical ontologies. *Methods of Information in Medicine*, 45:267–274.

[Rosse and Mejino, 2003] Rosse, C. and Mejino, J. L. (2003). A reference ontology for biomedical informatics: the Foundational Model of Anatomy. *Journal of Biomedical Informatics*, 36(6):478–500.

[Ruiz-Martínez, 2011] Ruiz-Martínez, J. M. (2011). *Metodología para la población automática de ontologías. Aplicación en los dominios de medicina y turismo*. PhD thesis, Universidad de Murcia.

[Ruttenberg et al., 2007] Ruttenberg, A., Clark, T., Bug, W., Samwald, M., Bodenreider, O., Chen, H., Doherty, D., Forsberg, K., Gao, Y., Kashyap, V., Kinoshita, J., Luciano, J., Marshall, M. S., Ogbuji, C., Rees, J., Stephens, S., Wong, G. T., Wu, E., Zaccagnini, D., Hongsermeier, T., Neumann, E., Herman, I., and Cheung, K.-H. (2007). Advancing translational research with the Semantic Web. *BMC Bioinformatics*, 8(Suppl 3):S2.

[Sanger, 1959] Sanger, F. (1959). Chemistry of Insulin: Determination of the structure of insulin opens the way to greater understanding of life processes. *Science*, 129(3359):1340–1344.

[Schmidt-Schau and Smolka, 1991] Schmidt-Schau, M. and Smolka, G. (1991). Attributive concept descriptions with complements. *Artificial Intelligence*, 48(1):1–26.

[Schober et al., 2009] Schober, D., Smith, B., Lewis, S. E., Kusnierczyk, W., Lomax, J., Mungall, C., Taylor, C. F., Rocca-Serra, P., and Sansone, S.-A. (2009). Survey-based naming conventions for use in OBO Foundry ontology development. *BMC Bioinformatics*, 10(1):125.

[Schriml et al., 2012] Schriml, L. M., Arze, C., Nadendla, S., Chang, Y.-W. W., Mazaitis, M., Felix, V., Feng, G., and Kibbe, W. a. (2012). Disease Ontology: a backbone for disease semantic integration. *Nucleic Acids Research*, 40(D1):D940–D946.

[Shadbolt et al., 2006] Shadbolt, N., Hall, W., and Berners-Lee, T. (2006). The semantic web revisited. *Intelligent Systems, IEEE*, 21(3):96–101.

[Shearer et al., 2008] Shearer, R., Motik, B., and Horrocks, I. (2008). HermiT : A Highly-Efficient OWL Reasoner. *Complexity*, 432:10.

[Shvaiko and Euzenat, 2013] Shvaiko, P. and Euzenat, J. (2013). Ontology Matching: State of the Art and Future Challenges. *IEEE Transactions on Knowledge and Data Engineering*, 25(1):158–176.

[Singh and Huhns, 2004] Singh, M. P. and Huhns, M. N. (2004). *Service-Oriented Computing*. John Wiley & Sons, Ltd, Chichester, UK.

[Sirin et al., 2007] Sirin, E., Parsia, B., Grau, B. C., Kalyanpur, A., and Katz, Y. (2007). Pellet: A practical OWL-DL reasoner. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2):51–53.

[Slater et al., 2015] Slater, L., Gkoutos, G., Schoeld, P. N., and Hoehndorf, R. (2015). Using Aber-OWL for fast and scalable reasoning over BioPortal ontologies. In *Proceedings of the ICBO 2015 International Conference on Biomedical Ontology 2005*, pages 81–76, Lisbon.

[Smith, 2003] Smith, B. (2003). Ontology. *The Blackwell Guide to the Philosophy of Computing and Information*, pages 153–166.

[Smith, 2004] Smith, B. (2004). Beyond concepts: ontology as reality representation. *Formal Ontology in Information Systems. IOS Press*, (November):4–6.

[Smith, 2009] Smith, B. (2009). An Introduction to Ontology: From Aristotle to the Universal Core (http://ontology.buffalo.edu/smith/IntroOntology_Course.html).

[Smith et al., 2007] Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., Eilbeck, K., Ireland, A., Mungall, C. J., Leontis, N., Rocca-Serra, P., Ruttenberg, A., Sansone, S.-A., Scheuermann, R. H., Shah, N., Whetzel, P. L., and Lewis, S. (2007). The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology*, 25(11):1251–1255.

[Smith et al., 2005] Smith, B., Ceusters, W., Klagges, B., Köhler, J., Kumar, A., Lomax, J., Mungall, C., Neuhaus, F., Rector, A. L., and Rosse, C. (2005). Relations in bioimedical ontologies. *Genome Biology*, 6(5):R46.

[Stephan et al., 2007] Stephan, G., Pascal, H., and Andreas, A. (2007). Knowledge Representation and Ontologies. In *Semantic Web Services*, pages 51–105. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Stevens et al., 2000] Stevens, R., Goble, C. a., and Bechhofer, S. (2000). Ontology-based knowledge representation for bioinformatics. *Briefings in bioinformatics*, 1(4):398–414.

[Stevens and Lord, 2009] Stevens, R. and Lord, P. (2009). Application of Ontologies in Bioinformatics. *Handbook on Ontologies*, pages 735–756.

[Stroetman et al., 2009] Stroetman, V., Kalra, D., Lewalle, P., Rector, A., Rodrigues, J., Stroetman, K., Surjan, G., Ustun, B., Virtanen, M., and Zanstra, P. (2009). Semantic Interoperability for Better health and Safer Healthcare [34 pages]. (January).

[Studer et al., 1998] Studer, R., Benjamins, V., and Fensel, D. (1998). Knowledge engineering: Principles and methods.

[Suárez-Figueroa et al., 2012] Suárez-Figueroa, M. C., Gómez-Pérez, A., and Fernández-López, M. (2012). The NeOn Methodology for Ontology Engineering. In *Ontology Engineering in a Networked World*, pages 9–34. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Tartir et al., 2005] Tartir, S., Arpinar, I., Moore, M., Sheth, a., and Aleman-Meza, B. (2005). OntoQA: Metric-Based Ontology Quality Analysis. *IEEE Workshop on Know-*

ledge Acquisition from Distributed, Autonomous, Semantically Heterogeneous Data and Knowledge Sources, pages 45–53.

[The International Health Terminology Standards Development Organisation, 2015] The International Health Terminology Standards Development Organisation (2015). SNOMED CT Document Library (http://ihtsdo.org/fileadmin/user_upload/doc/).

[Third, 2012] Third, A. (2012). "Hidden semantics": what can we learn from the names in an ontology? In *Proceedings of the Seventh International Natural Language Generation Conference*, Utica, IL, USA.

[Tim, Lee et al., 2001] Tim, Lee, B., Hendler, J., and Lassila, O. (2001). The Semantic Web. *Scientific American*, 284(5):28—-37.

[Tsarkov and Horrocks, 2006] Tsarkov, D. and Horrocks, I. (2006). FaCT++ Description Logic Reasoner: System Description. *Proceedings of the Third International Joint Conference (IJCAR 2006)*, pages 292–297.

[Uschold and King, 1995] Uschold, M. and King, M. (1995). Towards a methodology for buiding ontologies. In *IJCAI-95 Wokshop on Basic Ontological Issues in KNowledge Sharing.*

[Uzuner et al., 2010] Uzuner, O., Solti, I., and Cadag, E. (2010). Extracting medication information from clinical text. *Journal of the American Medical Informatics Association: JAMIA*, 17(5):514–518.

[Verspoor et al., 2009] Verspoor, K., Dvorkin, D., Cohen, K. B., and Hunter, L. (2009). Ontology quality assurance through analysis of term transformations. *Bioinformatics*, 25(2004):77–84.

[Whetzel et al., 2011a] Whetzel, P. L., Noy, N., Shah, N., Alexander, P., Dorf, M., Fergerson, R., Storey, M. A., Smith, B., Chute, C., and Musen, M. (2011a). BioPortal: Ontologies and integrated data resources at the click of a mouse. *CEUR Workshop Proceedings*, 833:292–293.

[Whetzel et al., 2011b] Whetzel, P. L., Noy, N. F., Shah, N. H., Alexander, P. R., Nyulas, C., Tudorache, T., and Musen, M. A. (2011b). BioPortal: enhanced functionality via new Web services from the National Center for Biomedical Ontology to access and use ontologies in software applications. *Nucleic acids research*, 39(Web Server issue):W541–5.

[Widdowson, 2007] Widdowson, H. (2007). J.R. Firth, 1957, Papers in Linguistics 1934–51. *International Journal of Applied Linguistics*, 17(3):402–413.

[Wroe et al., 2003] Wroe, C. J., Stevens, R., Goble, C. a., and Ashburner, M. (2003). A methodology to migrate the gene ontology to a description logic environment using DAML+OIL. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, 635:624–635.

[Yu, 2006] Yu, A. C. (2006). Methods in Biomedical Ontology. *Journal of Biomedical Informatics*, 39(3):252–266.

Apéndice A

Complexity analysis of the graph

ALGORITHMS FOR FINDING LEXICAL REGULARIES AND CPE-METRIC

In this appendix we show the algorithms used for creating the graph, searching the lexical regularitites, use the graph for seach if a group of tokens has an exact match in an ontology and calculate the CPE-Metric.

1. Load Ontology and build graph of labels – (Lines 1-19) and Search the whole set of LRs in an ontology – (Lines 21-25) ……………………………………………………
2. Search an LR of length L from a Node – (algorithm 2):……………………………
3. Using the graph for search if an string has an exact match in an ontology…
4. Using the graph for search decompositions ……………………………………………
    a. Finding decomposition Condition 1 ………………………………………..…….
    b. Finding decomposition Condition 2 ……………………………………………..….
    c. Finding decomposition Condition 3: …………………………………………….….

1. Load Ontology and build graph of labels – (Lines 1-19) and Search the whole set of LRs in an ontology – (Lines 21-25):

```
Function: SearchWholeSetOfLRs
Input:
    (1) ONT: OWL or OBO ontology file
    (2) CV:  Coverage Threshold
Output:
    (1) LRSET:  set with the lexical regularities (LRs) found
```

```
    1.  Load ONT in memory using a library for manipulating ontologies
    2.  FOR each CLASS in ONT
    3.    Extract the LABEL associated with CLASS
    4.    FOR each TOKEN of the LABEL
    5.      Search in the graph the node TOKEN
    6.      IF ( TOKEN not exists in the graph of labels )
    7.        Create NODE with id TOKEN
    8.        ADD NODE in a global HASHTABLE for query tokens in O(1)
    9.      END IF
    10.     IF ( TOKEN is not first token in LABEL )
    11.       Search in the graph the node TOKEN_PREC that precedes TOKEN
    12.       IF( ARROW not existes from TOKEN to TOKEN_PREC )
    13.         Create an ARROW from TOKEN to TOKEN_PREC
    14.     END IF
    15.     Register LABEL in the edge
    16.     END IF
    17.   END FOR
    18.   MNT = update the maximum number of tokens according to LABEL
    19. END FOR
    20.
    21. FOR each NODE in the graph of labels
    22.   FOR LR_LENGTH 1 TO MNT-1
    23.     LRSET = LRSET Union SearchLRs(NODE, NULL, LR_LENGTH-1, CV)
    24.   END FOR
    25. END FOR
```

2. Search an LR of length L from a Node – (algorithm 2):

```
Function: SearchLRs
Input:
    (1) NODE:   node to expand searching lexical regularities
    (2) ACS:    active class (ACS) set of identifiers
    (3) LENGTH: the remainder length of the lexical regularity
    (4) CV:     minimum coverage threshold of the lexical regularities
Output:
    (1) LRSET:  set with the lexical regularities (LRs) found
```

```
IF ( ACS id EMPTY ) RETURN ACS
IF ( |ACS| < CV ) LRSET = {}, RETURN LRSET
IF ( LENGTH is 0 )
  LRSET = ADD LR with ACS as exhibited classes, RETURN LRSET
END IF

FOR each ARROW departing from NODE
  NEXT_NODE = node where ARROW arrive
  IF ( ACS is NULL )
    ADD all the labels id register in ARROW to ACS
  END IF

  FOR each ARROW_EXP departing from NODE
    ADD all the labels ids register in ARROW_EXP to ACS_EXP
  END FOR

  ACS = ACS intersection ACS_EXP
  LRSET = LRSET Union SearchLRs(NEXT_NODE, ACS, LENGTH-1, CV)
  ACS is set as the initial value of the parameter
ENDFOR

RETURN LRSET
```

BASE CASES AND PRUNES

3. *Using the graph for search if an string has an exact match in an ontology*

```
Function: QueryLabelInOntology
Input:
    (1) LABEL_STR: string with the label to search in the ontology
    (2) GraphedOntology: graph with the ontology labels processed
Output:
    (1) ONTOLOGYO_CLASS:  return the ontology class that has LabelStr or NULL otherwise
```

```
TOKENS_LIST = obtain TOKENS from LABEL_STR
FIRST_TOKEN = first token in the TOKENS_LIST
LAST_TOKEN  = last token in TOKEN_LIST

NODE = search in the graph the node FIRST_TOKEN

LR = SearchLRs(NODE, ACS, |TOKENS_LIST|-1, CV=1)

IF LR is not NULL
  FOR each LR_LABEL that exhibits LR
    LABEL_TOKENS_LIST = obtain TOKENS from LR_LABEL
    IF ( |TOKENS_LIST| == |LABEL_TOKENS_LIST| )
      IF TOKENS_LIST is equal to LABEL_TOKENS_LIST
        RETURN identifier of LR_LABEL
      END IF
    END IF
  END FOR
END IF

RETURN NULL
```

4. Using the graph for search decompositions

a) Finding decomposition Condition 1

```
Function: FindDecomposition_CPEc1
Input:
    (1) LABEL_STR:   label to decompose
    (2) GraphOntologyE: graph with the ontologyE labels processed
Output:
    (1) Decompositions: set of tokens that are found in OntologyE as full labels
```

```
DECOMPOSITIONS_IN_ONTOLOGY_E = create empty set

FOR each TOKEN of the LABEL_STR
  IF QueryLabelInOntology(TOKEN, GraphOntologyE)
    Add DECOMPOSITIONS_IN_ONTOLOGY_E
  END IF
ENDFOR
RETURN DECOMPOSITIONS_IN_ONTOLOGY_E
```

b) Finding decomposition Condition 2

```
Function: FindDecomposition_CPEc2
Input:
    (1) LABEL_STR:   label to decompose
    (2) GraphOntologyE: graph with the ontologyE labels processed
Output:
    (1) Decompositions: set of tokens that are found in OntologyE as full labels
```

```
DECOMPOSITIONS_IN_ONTOLOGY_E = create empty set

SUB_TOKEN_LISTS = combination of consecutive tokens in LABELS_STR

FOR each SUB_TOKEN_LIST of the SUB_TOKEN_LISTS
  IF QueryLabelInOntology(SUB_TOKEN_LIST, GraphOntologyE)
    Add DECOMPOSITIONS_IN_ONTOLOGY_E
  END IF
ENDFOR
RETURN DECOMPOSITIONS_IN_ONTOLOGY_E
```

c) Finding decomposition Condition 3:

```
Function: FindDecomposition_CPEc3
Input:
    (1) LABEL_STR:   label to decompose
    (2) GraphOntologyS: graph with the ontologyS labels processed
    (3) GraphOntologyE: graph with the ontologyE labels processed
Output:
    (1) Decompositions: set of tokens that are found in OntologyE as full labels
```

```
DECOMPOSITIONS_IN_ONTOLOGY_S = create empty set
DECOMPOSITIONS_IN_ONTOLOGY_E = create empty set

SUB_TOKEN_LISTS = combination of consecutive tokens in LABELS_STR

FOR each SUB_TOKEN_LIST of the SUB_TOKEN_LISTS
  IF QueryLabelInOntology(SUB_TOKEN_LIST, GraphOntologyS)
    Add DECOMPOSITIONS_IN_ONTOLOGY_S
    Mark tokens indexes of SUB_TOKEN_LIST as matched
  END IF
END FOR

FOR each SUB_TOKEN_LIST of the SUB_TOKEN_LISTS
```

```
  IF QueryLabelInOntology(SUB_TOKEN_LIST, GraphOntologyE)
    Add DECOMPOSITIONS_IN_ONTOLOGY_E
    Mark tokens indexes of SUB_TOKEN_LIST as matched
  END IF
ENDFOR

IF all indexes are marked as matched
  RETURN DECOMPOSITIONS_IN_ONTOLOGY_S
              and DECOMPOSITIONS_IN_ONTOLOGY_E
END IF

RETURN NULL
```

COMPLEXITY ANALYSIS OF THE ALGORITHMS FOR FINDING LEXICAL REGULARIES AND CPE-METRIC

In this appendix we show the algorithms used for doing the experiments. We calculate its complexity. Due to the space restriction in the paper, we just included in the paper the algorithm for creating the graph. And discuss the benefits of using the graph in terms of Big-O.

1. Load Ontology and build graph of labels – (algorithm 1, Lines 1-19):…………..
2. Search an LR of length L from a Node – (algorithm 2):……………………………….
3. Search the whole set of LRs in an ontology – (algorithm 1, Lines 21-25):………
4. Using the graph for search a class in an ontology:……………………………………….
5. Using the graph for search decompositions: ………………………………………………..
    a. Finding decomposition Condition 1: ………………………………………………..
    b. Finding decomposition Condition 2: ………………………………………………...
    c. Finding decomposition Condition 3: ………………………………………………..

Summary of the execution times:

Description of each variable that has influence in the execution time

t = number of unique tokens tokens
tl = max number of tokens in labels
n = length of the regularity
l = max num. of labels with two consecutive repeated tokens
c = max. number of classes
a = max number of arrows that depart from nodes

Load Ontology and build graph of labels – (algorithm 1, Lines 1-19):
$t1+ t2*2O(c)+10\ O(c*tl)+O(c)$
Search an LR of length L from a Node – (algorithm 2):
$O(n*a^2*l)+ 2\ O(n*a*l^2) + 3\ O(n*a)$
Search the whole set of LRs in an ontology – (algorithm 1, Lines 21-25):
$O(t*tl*n*a^2)+ 2\ O(t*tl*n*a*l^2) + O(t*tl*n*a) + O(t*tl)$
Using the graph for search a class in an ontology:
$O(tl)+5\ O(1)+O(n*a^2*l)+O(n*a*l^2)+O(n*a)+2\ O(c*tl)+2\ O(c)$
Finding decomposition Condition 1:
$O(1)+O(tl)*((tl)+5\ O(1)+O(n*a^2*l)+O(n*a*l^2)+O(n*a)+2\ O(c*tl)+2\ O(c))+O(1)$
Finding decomposition Condition 2:
$2\ O(1)+(tl^3)+5\ O(tl^2)+O(n*a^2*l*tl^2)+O(n*a*l^2*tl^2)+O(n*a*tl^2)+2\ O(c*tl^3)+2\ O(c*tl^2))+O(tl^2)$
Finding decomposition Condition 3:
$O(tl^2)$ $+$ $(tl^3)+5\ O(tl^2)+O(n*a^2*l*tl^2)+O(n*a*l^2*tl^2)+O(n*a*tl^2)+2\ O(c*tl^3)+2\ O(c*tl^2)$ $+$ $(tl^3)+5\ O(tl^2)+O(n*a^2*l*tl^2)+O(n*a*l^2*tl^2)+O(n*a*tl^2)+2\ O(c*tl^3)+2\ O(c*tl^2)$ $+$ $O(tl)$ $+$ $O(1)$

Load Ontology and build graph of labels – (algorithm 1, Lines 1-19):

```
1.  Load ONT in memory using a library for manipulating ontologies          $t_1$
    This time depend on the library for manipulating the ontology
2.  FOR each CLASS in ONT                                                   O(c)
    "c" is the max. number of classes
3.    Extract the LABEL associated with CLASS                               $t_2$
      This time depend on the library for manipulating the ontology
4.    FOR each TOKEN of the LABEL                                           O(tl)
      "tl" is the max number of tokens in labels
5.      Search in the graph the node TOKEN                                  O(ht-Q-t)
        Tokens are indexes in a hash-table for faster individual queries
        $T_{ex}$ depends of the hash-table used. Query time
6.      IF ( TOKEN not exists in the graph of labels )                      O(1)
          Constant time, allocate memory for the node
7.        Create NODE with id TOKEN                                         O(1)
          Constant time, allocate memory for the node
8.        ADD NODE in a global HASHTABLE for query tokens in O(1)           O(ht-I-t)
          $T_{ex}$ depends of the hash-table used. Insertion time
9.      END IF
10.     IF ( TOKEN is not first token in LABEL )                            O(1)
          Constant time, check index
11.       Search in the graph the node TOKEN_PREC that precedes TOKEN       O(ht-Q-t)
          $T_{ex}$ depends of the hash-table used. Query time
12.       IF( ARROW not existes from TOKEN to TOKEN_PREC )                  O(ht-Q-e)
          $T_{ex}$ depends of the hash-table used. Query time
          Arrows are stored in hash-table which is indexed by TOKEN_PREC
13.         Create an ARROW from TOKEN to TOKEN_PREC                        O(1)
            Constant time, check index
14.     END IF
15.     Register LABEL in the edge                                          O(ht-I-e)
        Create edge. $T_{ex}$ depends of the hash-table used. Insertion time
16.     END IF
17.   END FOR
18.   MNT = update the maximum number of tokens according to LABEL          O(1)
      Constant time, allocate memory for the node
19. END FOR
```

COMPLEXITY:

```
$t_1$+O(c)*(
           $t_2$+O(tl)*(
                O(ht-Q-t)+O(1)+ O(1)+O(ht-I-t)+O(1)+O(ht-Q-t)+O(ht-Q-e)+O(1)+O(ht-I-e)
                    )+O(1)
      )
```

```
$t_1$+O(c)*(
           $t_2$+O(tl)*(
                2 O(ht-Q-t) + 4 O(1) + O(ht-I-t) + O(ht-Q-e) + O(ht-Q-e) + O(ht-I-e)
                    )+O(1)
      )
```

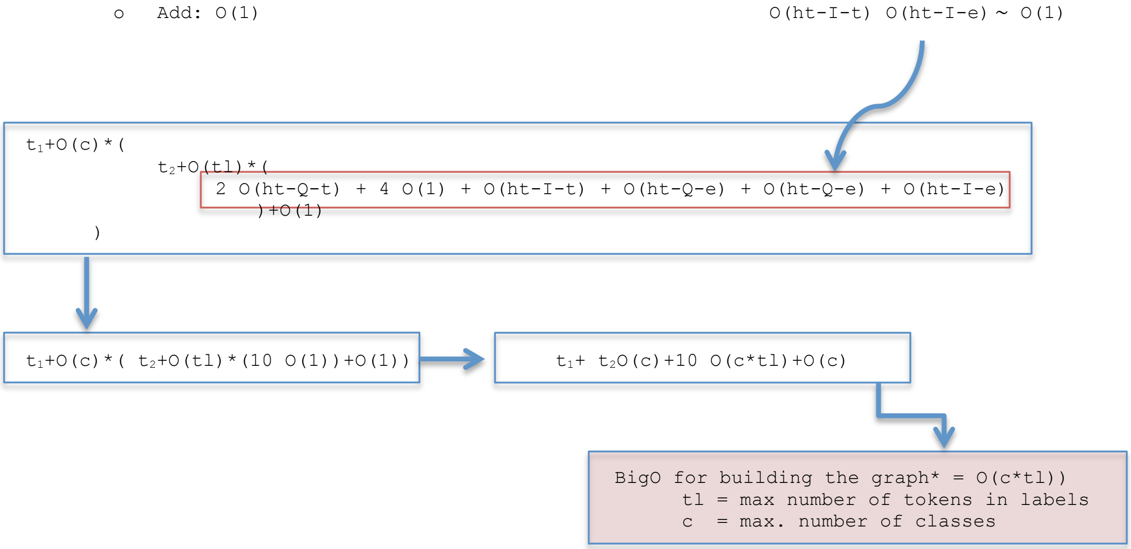
Our algorithm is implemented in Java and we used the collections implemented in the SDK 1.6. We have found the complexity of the operations commented in next link:

http://www.javaexperience.com/time-complexity-of-collection-classes/

- ***HashMap time complexity***: The elements are placed randomly as per the hashcode. Here the assumption is that a good implementation of hashcode has been provided.
    - Read/Search any element: O(1)

O(ht-Q-t) O(ht-Q-e)~ O(1)

- ○ Update: O(1)
- ○ Delete: O(1)
- ○ Add: O(1)



```
O(ht-I-t) O(ht-I-e) ~ O(1)

t₁+O(c)*(
        t₂+O(tl)*(
            2 O(ht-Q-t) + 4 O(1) + O(ht-I-t) + O(ht-Q-e) + O(ht-Q-e) + O(ht-I-e)
               )+O(1)
    )

t₁+O(c)*( t₂+O(tl)*(10 O(1))+O(1))

t₁+ t₂O(c)+10 O(c*tl)+O(c)

BigO for building the graph* = O(c*tl))
     tl = max number of tokens in labels
     c  = max. number of classes
```

*Assuming a well balanced HashMap

Search an LR of length L from a Node – (algorithm 2):

```
IF ( ACS id EMPTY ) RETURN ACS                                          O(1)
Tex depends of the hash-set used. Empty operation is O(1)
IF ( |ACS| < CV  ) LRSET = { }, RETURN LRSET                            O(1)
Tex depends of the hash-set used. Size operation is O(1)      BASE CASES
IF ( LENGTH is 0  )                                           AND PRUNES O(1)
Constant time, comparation
  LRSET = ADD LR with ACS as exhibited classes, RETURN LRSET             O(1)
  Constant time, create LR an associate set
END IF

FOR each ARROW departing from NODE                                      O(a)
"a" is the max number of arrows that depart from nodes
  NEXT_NODE = node where ARROW arrive                                   O(ht-Q-t)
  Tex depends of the hash-table used. Insertion time
  IF ( ACS is NULL )                                                    O(1)
  Constant time, comparison
    ADD all the labels id register in ARROW to ACS                      O(hs-D-l)
    "l" is the max num. of labels with two consecutive tokens repeated
    Tex depends of the hash-set used. Duplication
  END IF

  FOR each ARROW_EXP departing from NODE                                O(a)
  "a" is the max number of arrows that depart from nodes
    ADD all the labels id register in ARROW_EXP to ACS_EXP              O(hs-D-l)
    "l" is the max num. of labels with two consecutive tokens repeated
    Tex depends on the hash-set used. Duplication
  END FOR

  ACS = ACS intersection ACS_EXP                                        O(hs-∩-l)
  Tex depends on the hash-set used. Intersection
  LRSET = LRSET Union SearchLRs(NEXT_NODE, ACS, LENGTH-1, CV)         ¿R+O(hs-U-l)?
  Tex depends on the hash-set used. Recursivity + Union
  ACS is set as the initical value of the parameter                     O(1)
  Constant time, assignation
ENDFOR

RETURN LRSET
```

COMPLEXITY:

This function is recursive so next we calculate the BigO of it. The parameters that reduce de function is the third parameters LENGTH.

$$t(n)\begin{cases} t(1) = 1 \\ t(n)=(O(a)*(\ O(ht\text{-}Q\text{-}t)+\ O(1)+\ O(hs\text{-}D\text{-}l)+\ (O(a)*\ O(hs\text{-}D\text{-}l))+\ O(hs\text{-}\cap\text{-}l)+\ O(hs\text{-}U\text{-}l)))*t(n-1) \end{cases}$$

- ***HashSet time complexity***: The elements are distributed randomly in memory using their hashcode. Here also the assumption is that good hashcode which generated unique hashcode for different objects has been provided.
  - Read/Search any element O(1)
  - Update : O(1)
  - Delete : O(1)
  - Add : O(1)

Then:
- Duplicate: O(hs-D) = O(n)
  n is the size of the set and add operation is O(1)
- O(hs-∩) = O(n*m)
  "n" is the size of the first set and "m" the size of the second set
- Union: O(hs-U) = O(n*m)
  "n" is the size of the first set and "m" the size of the second

Substituting in the t(n) formula:

$$t(n)\begin{cases} t(1) = 1 \\ t(n)=(O(a)*(\ O(1)+ O(1)+ O(l)+ (O(a)* O(l))+ O(l^2)+ O(l^2)))*t(n-1) \end{cases}$$

$$t(n)\begin{cases} t(1) = 1 \\ t(n)=(O(a)*(\ 3\ O(1) + O(a*l)+ 2\ O(l^2)))*t(n-1) \end{cases}$$

$$t(n)\begin{cases} t(1) = 1 \\ t(n)=(\ 3\ O(a) + O(a^2*l)+ 2\ O(a*l^2))*t(n-1) \end{cases}$$

$$t(n) = O(n*a^2*l)+ 2\ O(n*a*l^2) + 3\ O(n*a)$$

BigO to find a regularity of length “n” = $O(n*a^2*l)+O(n*a*l^2)+O(n*a)$
n = length of the regularity
a = máximum number of arrows that depart from nodes
l = max num. of labels with two consecutive tokens repeated

Search the whole set of LRs in an ontology – (algorithm 1, Lines 21-25):

```


FOR each NODE in the graph of labels                                        O(t)
  "t" is number of unique tokens tokens
  FOR LR_LENGTH 1 TO MNT-1                                                  O(tl)
   "tl" is the max number of tokens in labels
    LRSET = LRSET Union SearchLRs(NODE, NULL, LR_LENGTH-1, CV)          ¿Call+O(hs-U-lr)?
     Tex depends of the hashset used. Call SearchLRs + Union of LRs
  END FOR
END FOR
```

$$t(n) = O(t)*O(tl)*(O(n*a^{2})+ 2 O(n*a*l^{2}) + O(n*a) + O(hs-U-lr))$$

$$t(n) = O(t)*O(tl)*(O(n*a^{2})+ 2 O(n*a*l^{2}) + O(n*a) + O(1))$$

$$t(n) = O(t)*(O(tl*n*a^{2})+ 2 O(tl*n*a*l^{2}) + O(tl*n*a) + O(tl))$$

$$t(n) = O(t*tl*n*a^{2})+ 2 O(t*tl*n*a*l^{2}) + O(t*tl*n*a) + O(t*tl)$$

BigO to find a the whole set of lexical regularities =
$O(l*tl*n*a^{2})+ 2 O(l*tl*n*a*l^{2}) + O(l*tl*n*a) + O(l*tl)$

t = number of unique tokens tokens
tl = max number of tokens in labels
n = length of the regularity
a = máximum number of arrows that depart from nodes
l = max num. of labels with two consecutive tokens repeated

Using the graph for search a class in an ontology:

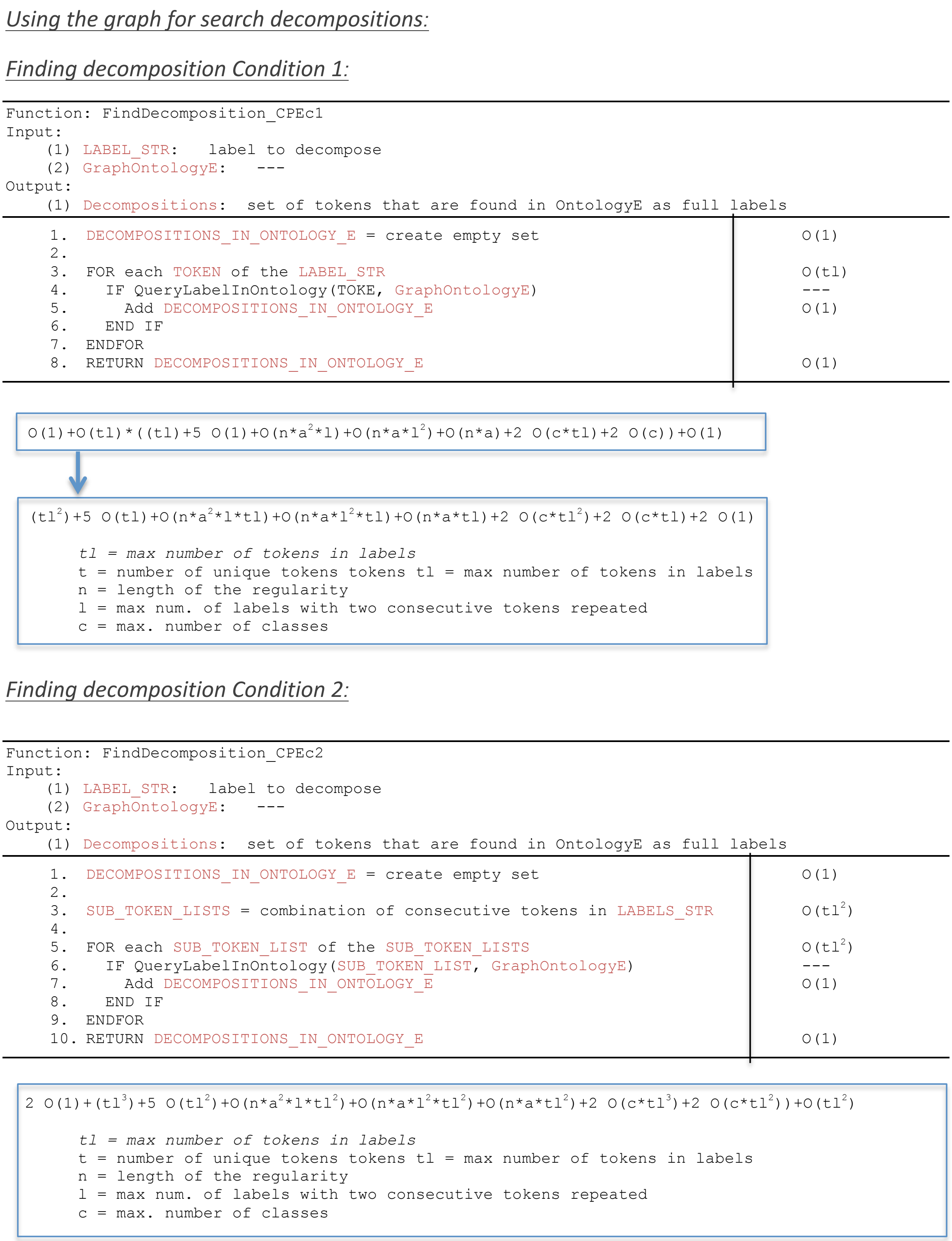
```
Function: QueryLabelInOntology
Input:
    (1) LABEL_STR: string with the label to search in the ontology
    (2) GraphedOntology: graph with the ontology labels processed
Output:
    (1) ONTOLOGYO_CLASS:  return the ontology class that has LabelStr or NULL otherwise

TOKENS_LIST = obtain TOKENS from LABEL_STR                            O(tl)
FIRST_TOKEN = first token in the TOKENS_LIST                          O(1)
LAST_TOKEN  = last token in TOKEN_LIST                                O(1)

NODE = search in the graph the node FIRST_TOKEN                       O(ht-Q-t)

LR = SearchLRs(NODE, ACS, |TOKENS_LIST|-1, CV=1)                      $O(n*a^2*l)+O(n*a*l^2)$
                                                                      +O(n*a)
IF LR is not NULL                                                     O(1)
  FOR each LR_LABEL that exhibits LR                                  O(c)
    LABEL_TOKENS_LIST = obtain TOKENS from LR_LABEL                   O(tl)
    IF ( |TOKENS_LIST| == |LABEL_TOKENS_LIST| )                       O(1)
      IF TOKENS_LIST is equal to LABEL_TOKENS_LIST                    O(tl)
        RETURN identifier of LR_LABEL                                 O(1)
      END IF
    END IF
  END FOR
END IF

RETURN NULL                                                           O(1)
```

O(tl)+O(1)+O(1)+O(ht-Q-t)+O(1)+$O(n*a^2*l)$+$O(n*a*l^2)$+O(n*a)+O(1)+O(c)*(O(tl)+O(1)+O(tl)+O(1))+O(1)

O(tl)+O(1)+O(1)+O(ht-Q-t)+O(1)+$O(n*a^2*l)$+$O(n*a*l^2)$+O(n*a)+O(1)+O(c*tl)+O(c)+O(c*tl)+O(c)+O(1)

O(tl)+5 O(1)+$O(n*a^2*l)$+$O(n*a*l^2)$+O(n*a)+2 O(c*tl)+2 O(c)

tl = max number of tokens in labels
t = number of unique tokens tokens tl = max number of tokens in labels
n = length of the regularity
l = max num. of labels with two consecutive tokens repeated
c = max. number of classes



Using the graph for search decompositions:

Finding decomposition Condition 1:

```
Function: FindDecomposition_CPEc1
Input:
    (1) LABEL_STR:   label to decompose
    (2) GraphOntologyE:   ---
Output:
    (1) Decompositions:  set of tokens that are found in OntologyE as full labels
```

```
DECOMPOSITIONS_IN_ONTOLOGY_E = create empty set                  O(1)

FOR each TOKEN of the LABEL_STR                                  O(tl)
  IF QueryLabelInOntology(TOKE, GraphOntologyE)                  ---
    Add DECOMPOSITIONS_IN_ONTOLOGY_E                             O(1)
  END IF
ENDFOR
RETURN DECOMPOSITIONS_IN_ONTOLOGY_E                              O(1)
```

$O(1)+O(tl)*((tl)+5\ O(1)+O(n*a^2*l)+O(n*a*l^2)+O(n*a)+2\ O(c*tl)+2\ O(c))+O(1)$

$(tl^2)+5\ O(tl)+O(n*a^2*l*tl)+O(n*a*l^2*tl)+O(n*a*tl)+2\ O(c*tl^2)+2\ O(c*tl)+2\ O(1)$

tl = max number of tokens in labels
t = number of unique tokens tokens tl = max number of tokens in labels
n = length of the regularity
l = max num. of labels with two consecutive tokens repeated
c = max. number of classes

Finding decomposition Condition 2:

```
Function: FindDecomposition_CPEc2
Input:
    (1) LABEL_STR:   label to decompose
    (2) GraphOntologyE:   ---
Output:
    (1) Decompositions:  set of tokens that are found in OntologyE as full labels
```

```
DECOMPOSITIONS_IN_ONTOLOGY_E = create empty set                  O(1)

SUB_TOKEN_LISTS = combination of consecutive tokens in LABELS_STR  $O(tl^2)$

FOR each SUB_TOKEN_LIST of the SUB_TOKEN_LISTS                   $O(tl^2)$
  IF QueryLabelInOntology(SUB_TOKEN_LIST, GraphOntologyE)        ---
    Add DECOMPOSITIONS_IN_ONTOLOGY_E                             O(1)
  END IF
ENDFOR
RETURN DECOMPOSITIONS_IN_ONTOLOGY_E                              O(1)
```

$2\ O(1)+(tl^3)+5\ O(tl^2)+O(n*a^2*l*tl^2)+O(n*a*l^2*tl^2)+O(n*a*tl^2)+2\ O(c*tl^3)+2\ O(c*tl^2))+O(tl^2)$

tl = max number of tokens in labels
t = number of unique tokens tokens tl = max number of tokens in labels
n = length of the regularity
l = max num. of labels with two consecutive tokens repeated
c = max. number of classes

Finding decomposition Condition 3:

```
Function: FindDecomposition_CPEc3
Input:
    (1) LABEL_STR:   label to decompose
    (2) GraphOntologyS:   ---
    (3) GraphOntologyE:   ---
Output:
    (1) Decompositions:  set of tokens that are found in OntologyE as full labels

DECOMPOSITIONS_IN_ONTOLOGY_S = create empty set
DECOMPOSITIONS_IN_ONTOLOGY_E = create empty set

SUB_TOKEN_LISTS = combination of consecutive tokens in LABELS_STR        $O(tl^2)$

FOR each SUB_TOKEN_LIST of the SUB_TOKEN_LISTS                           $O(tl^2)$
  IF QueryLabelInOntology(SUB_TOKEN_LIST, GraphOntologyS)                ---
    Add DECOMPOSITIONS_IN_ONTOLOGY_S                                     $O(1)$
    Mark tokens indexes of SUB_TOKEN_LIST as matched                     $O(tl)$
  END IF
END FOR

FOR each SUB_TOKEN_LIST of the SUB_TOKEN_LISTS                           $O(tl^2)$
  IF QueryLabelInOntology(SUB_TOKEN_LIST, GraphOntologyE)                ---
    Add DECOMPOSITIONS_IN_ONTOLOGY_E                                     $O(1)$
    Mark tokens indexes of SUB_TOKEN_LIST as matched                     $O(tl^2)$
  END IF
ENDFOR

IF all indexes are marked as matched                                     $O(tl)$
  RETURN DECOMPOSITIONS_IN_ONTOLOGY_S                                    $O(1)$
            and DECOMPOSITIONS_IN_ONTOLOGY_E
END IF

RETURN NULL
```

$O(tl^2)$
+
$(tl^3)+5\ O(tl^2)+O(n*a^2*l*tl^2)+O(n*a*l^2*tl^2)+O(n*a*tl^2)+2\ O(c*tl^3)+2\ O(c*tl^2)$
+
$(tl^3)+5\ O(tl^2)+O(n*a^2*l*tl^2)+O(n*a*l^2*tl^2)+O(n*a*tl^2)+2\ O(c*tl^3)+2\ O(c*tl^2)$
+
$O(tl)$
+
$O(1)$

tl = max number of tokens in labels
t = number of unique tokens tokens tl = max number of tokens in labels
n = length of the regularity
l = max num. of labels with two consecutive tokens repeated
c = max. number of classes