Master in Artificial Intelligence (UPC-URV-UB)

Master of Science Thesis

Ontology-based Information Extraction

Carlos Vicient Monllaó

Advisors: Antonio Moreno Ribas, David Sánchez Ruenes

June, 23rd 2011

Agraïments

Aquest treball ha rebut el suport d'una beca predoctoral finançada per la Universitat Rovira i Virgili dintre del grup de recerca *Intelligent Technologies for Advanced Knowledge Adquisition* (ITAKA).

Vull aprofitar aquestes línies per agrair als meus directors, el Dr. Antonio Moreno i el Dr. David Sánchez, tota l'ajuda, els consells i les directrius que m'han proporcionat durant la realització d'aquest treball de màster.

Tampoc vull oblidar els meus companys del grup de recerca i, sobretot, la meva família, que no ha deixat d'animar-me a continuar amb els meus estudis.

Table of contents

List of figures

List of tables

1 Introduction

Since the creation of the World Wide Web (referred as WWW), presented by Tim Berners-Lee in 1989, its structure and architecture have been in constant growth and development. Nowadays the Web is involved in what we know as the Social Web or Web 2.0, where the user role changes and he becomes as consumer as producer of information, so, all users are able to add and modify their contents. This fact has as a result an exponential growth of the available contents. Although this increase of information could seem a very interesting feature, the lack of structure brought some problems: it complicates its accessing, and it cannot be interpreted semantically by IT applications(Fensel, Bussler et al. 2002), both manually and in an automatic way. So, in order to solve these inconveniences, the Semantic Web (Berners-Lee, Hendler et al. 2001) is proposed as a new global initiative.

The Semantic Web is an evolving extension of the World Wide Web in which the semantics of information and services on the Web is defined, making it possible for the Web to understand and satisfy the requests of people and machines to use its content. One of the basic pillars of the Semantic Web concept is the idea of having explicit semantic information on the Web pages that can be used by intelligent agents in order to solve complex problems of Information Retrieval and Question Answering. In consequence, the final objective of the Semantic Web is to be able to semantically analyze and catalog the Web contents. This requires a set of structures to model the knowledge, and a linkage between the knowledge and contents. In this manner the Semantic Web relies on two basic components, ontologies and semantic annotations.

Ontologies are formal, explicit specifications of a shared conceptualization. This means that ontologies are useful to model knowledge in a formal abstract way which can be read by computers. With ontologies it is possible to represent concepts, relations among concepts and even constraints on their use.

Annotations are a linkage between the knowledge and contents. On one hand, knowledge is represented by means of ontologies. On the other hand, contents are pieces of raw text that need a meaning and which are linked with ontological concepts.

Due to the interest in automated analysis of all this information, in recent years, there has been a growing interest in the research community in developing data

mining techniques, such as knowledge-based data mining and classification algorithms(Batet, Valls et al. 2010), which are able to exploit this kind of information. These methods rely on predefined knowledge (such as ontologies(Guarino 1998)) to semantically interpret textual data and extract more accurate conclusions from their analyses. They are typically applied over structured textual attributes which correspond to features of the analysed entities. In these cases, attribute labels (i.e., words or noun phrases) are interpreted by mapping them to concepts and analysing the background knowledge structure to which these concepts belong. However, these methods are rarely able to deal with raw text, from which relevant features should be extracted and matched to ontological entities before the data analysis. In this manner, textual documents (which represent most of available Web resources) describing a particular entity (e.g. questionnaires, Wikipedia entries, etc.) are difficult to process in order to extract relevant features which could be exploited in order to apply semantically focused data mining algorithms(Hotho, Maedche et al. 2002).

The main problem of Semantic Web is the fact that it is supposed that all Web contents are semantically annotated, and nowadays this is not true yet. As a result of those limitations, Semantic-based information extraction appears. It relies on ontologies in order to interpret the textual content of a resource regardless of its format. Even though there have been many conceptual approximations in the field of Semantic Web in which it is assumed that resources have been semantically annotated, in the short-term future it cannot be expected the availability of a massive amount of annotated Web resources. So, in order to take profit from the Web resources which are currently available, the extraction of features from plain text, as it is proposed in this work, goes through the syntactic analysis of its content and its association with the concepts modelled in one or more input ontologies.

To sum up, Semantic Web has brought about a growing interest in the research community in developing semantic data mining techniques. These techniques are able to exploit efficiently the semantic information but they depend on a structured input. Unfortunately, at the moment, most of available Web resources are in raw text. For all these reasons it is important to have mechanisms able to take profit of raw texts.

This work aims to ease the application of semantically-grounded data-mining algorithms on textual data and semi-structured resources.

1.1 DAMASK

Next, it is presented DAMASK, the project where this work is involved in. DAMASK means Data-mining algorithms with Semantic Knowledge and is a project founded by the Spanish Ministry of Science and Innovation and the Spanish Government (PlanE, Spanish Economy and Employment Stimulation Plan).

DAMASK proposes the use of semantic domain knowledge, represented in the form of ontologies, to define new methods for extracting and integrating information from heterogeneous Web resources with varying degrees of structure, performing an automatic classification, and making a semantic interpretation of the results.

The main DAMASK's goal is to develop new data mining methods driven by the semantic domain knowledge. More concretely, the project is centred in the application of ontologies to the following aspects:

1) Pre-processing of input data, focusing on their acquisition from freely and massively available resources such as Web resources, their integration and their transformation in a format which may be directly processed.
2) Methods of automatic classification of data, considering any type of heterogeneous information, including numerical, categorical and conceptual data.
3) Methods for interpreting the classes obtained in the previous step.

Concretely, the project is divided into 3 main tasks: ontology-based information extraction and integration from heterogeneous Web resources (Task 1), automatic clustering of entities based on the semantics of the concepts and attributes obtained from the Web resources (Task 2) and application of the developed methods to a Tourism test case (Task 3). In addition, another task is planned for management, dissemination and exploitation of the results of the project (Task 4).

Particularly, this work is involved in Task 1 of DAMASK which consists in designing a methodology for knowledge extraction from the Web. The information obtained is represented in a data matrix of object × attribute pairs which will be the input of Task 2. The data construction process is guided by the domain ontology (in OWL-DL). In this process, different levels are considered according to the structure of the resources: from ill-structured resources (e.g. Wikipedia), to non-structured textual Web pages. Finally, an integration process puts all the data obtained from each of these different sources into an heterogeneous data matrix, keeping, as much as possible, the knowledge regarding the objects that are being studied. Figure 1 shows the proposed methodology for task 1 and task 2 of DAMASK.

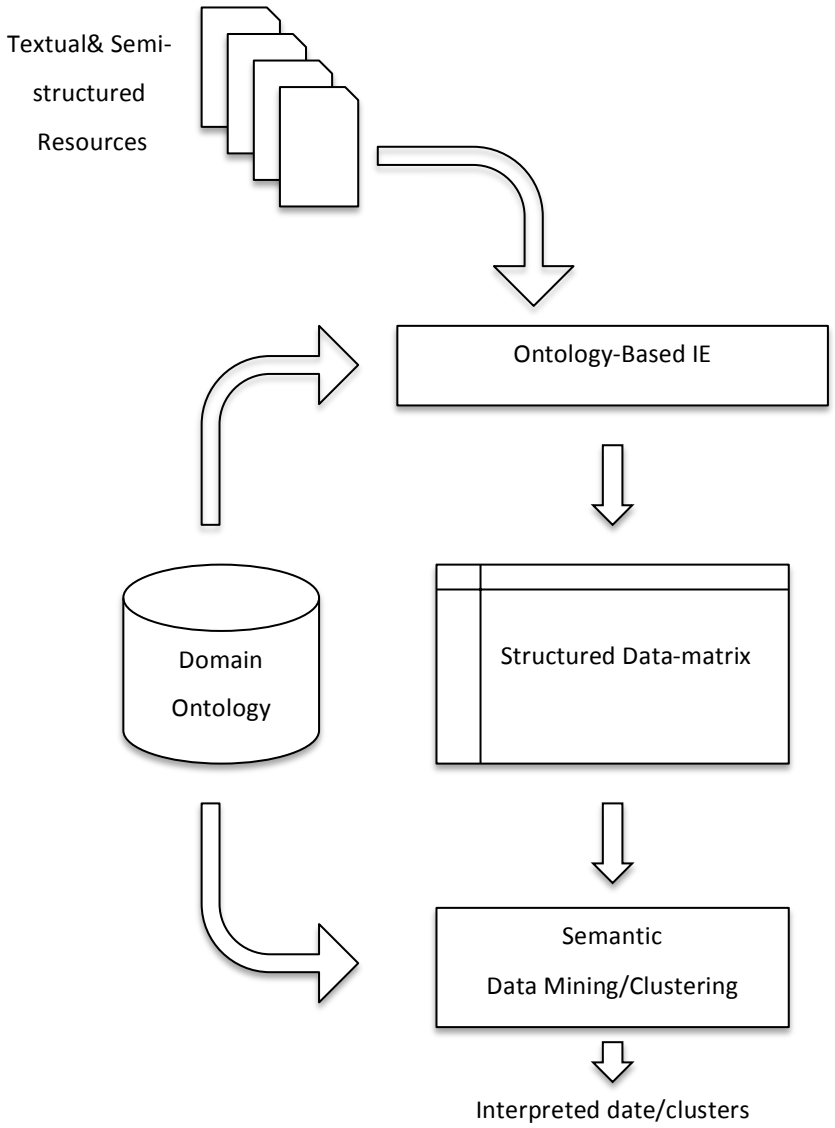


Figure 1 Proposed architecture

1.2 Objectives

The main goal of this work is to design and implement a novel method that is able to extract relevant features from a range of textual documents going from complete plain textual data to semi-structured. The designed methodology will be able to take profit from pre-processed input when it is available in order to complement its own learning algorithms. The key point of the work is to

complement the syntactical parsing and several natural language processing techniques with the knowledge contained in an input ontology (ideally, it should model the knowledge domain in which the posterior data analysis will be focused – e.g. touristic points of interest) in order to be able to:

1) identify relevant features describing a particular entity from textual data,
2) To associate, if applicable, extracted features to concepts contained in the input ontologies. In this manner, the output of the system would consist on tagged features which can be directly exploited by semantically grounded data mining algorithms (e.g. clustering) in order to classify them.

1.3 Overview of the methodology

The basic task of the work is to design a methodology that, taking raw text describing a certain entity as input will be able to:

a) Detect features describing or associated to the entity. This stage will focus on Named Entities (see section 4.1.2).
b) To assess which of the extracted features are more closely related to the entity (i.e. they better identify and describe it) in order to maximize the accuracy of the data mining process.
c) To associate the selected features to concepts modelled in an input ontology, if they fit in the domain covered by the ontology.

Steps (a) and (b) are focused on objective 1) and step (c) on objective 2 of section 1.2.

The whole process is unsupervised and automatic; thus, it is a scalable solution that can be applied regardless of the type of entities or the knowledge domain (i.e., it is domain independent) and which maximizes its generality and applicability(Sánchez 2008). Natural language related problems such as ambiguity are considered in order to improve the quality of the results. The scalability of the approach is also carefully considered, minimizing the dependency on external resources. Unsupervised learning techniques such as the use of statistical analyses evaluating information distribution (Turney 2001) and general linguistic patterns (Etzioni, Cafarella et al. 2005) can aid on this purpose.

1.4 Document structure

The rest of this document is organised as follows:

- Chapter 2 presents a state-of-the-art on Information Extraction and how to Ontologies have been used for Ontology-Based Information Extraction approaches.

- In chapter 3, all the learning techniques and tools used in this work are introduced. Moreover, it is discussed that the Web is a valid knowledge repository to support its use as the corpus for our work.
- Chapter 4 explains the proposed methodology for this work, which is unsupervised and domain independent. First it is presented a generic algorithm to explain how the methodology works and then, it is argued how to take profit from different types of Web resources.
- Chapter 5 includes the testing and evaluation of the extraction process explained in the previous chapter. A study of how the different parameters affect the final result of extracted features is also included.
- Chapter 6 summarises a list of conclusions of this work and devises some lines of future work. Moreover, the contributions of this work and publications are presented.

2 Ontology-based IE

This section discusses all related works in information extraction distinguishing algorithms to extract structured, semi-structured and non-structured resources.

This chapter is structured as follows:

- In §2.1, it is presented an overview of general Information Extraction and a comparison between traditional IE and Open IE is stated.
- §2.2 introduces the using of ontologies in Information Extraction approaches and discusses the importance of the extraction of semantic data.

2.1 Information Extraction

There has been an explosive growth in the amount of information available on networked computers around the world, much of it in the form of natural language documents. Information Extraction (IE) is the task of locating specific pieces of data within a natural language document (Xiao, Wissmann et al. 2004). Moreover, the advent of the internet has given IE a particular commercial relevance.

IE is a process which takes unseen texts as input and produces fixed format, unambiguous data as output. At the core of an IE system is an extractor, which processes text; it overlooks irrelevant words and phrases and attempts to home in on entities and the relationships between them (Etzioni, Banko et al. 2008). These data may be used directly for display to users, or may be stored in a database or spread sheet for direct integration with a back-office system, or may be used for indexing purposes in search engine/Information Retrieval (IR) applications (Xiao, Wissmann et al. 2004). If we compare IE and IR, whereas IR simply finds texts and presents them to the user (as classic search engines), IE analyses texts and presents only the specific information extracted from the text that is of interest to a user.

In the context of Web resources, a set of extraction rules suitable to extract information from a Web site is called a wrapper (Flesca, Manco et al. 2004). Two main approaches for wrapper generation tools have been proposed during the last

years: one is based on knowledge engineering –supervised, traditional IE– and the other on automatic training –unsupervised, open IE–. In the first, the domain expert has to manually design the extraction rules or tag some documents, which are used by an algorithm to obtain the appropriate extraction rules. In such an approach the user skills play a crucial role in the successful identification and analysis of relevant information. In the second, open IE exploits AI techniques to induce extraction rules starting from a set of generic information patterns. In Table 1, as stated in (Cimiano 2006) the main advantages and disadvantages of both approaches are summarised.

	Traditional IE	**Open IE**
Input	Corpus + Labelled Data	Corpus + Domain Independent Methods
Relations	Specified in advance	Discovered automatically
Complexity	O (D * R) D documents, R relations	O (D) D documents
Precision	Very precise (hand-coded rules)	Reasonable precision (rule induction)
Training	Expensive development & test cycle	Provide training data (expensive)
Patterns	Need to develop grammars	No need for developing grammars

Table 1 Comparison of traditional IE and Open IE

2.1.1 Traditional IE systems

Traditional methods on IE have focused on the use of supervised learning techniques such as hidden Markov models (Freitag and McCallum 1999; Skounakis, Craven et al. 2003), self-supervised methods (Etzioni, Cafarella et al. 2005), rule learning (Soderland 1999), and conditional random fields (McCallum 2003). These techniques learn a language model or a set of rules from a set of hand-tagged training documents and then apply the model or rules to new texts. Models learned in this manner are effective on documents similar to the set of training documents, but extract quite poorly when applied to documents with a different genre or style. As a result, this approach has difficulty scaling to the Web due to the diversity of text styles and genres on the Web and the prohibitive cost of creating an equally diverse set of hand tagged documents.

The most representative example of this kind of systems is KnowItAll (Etzioni, Cafarella et al. 2005). The KnowItAll Web IE system took the next step in automating IE by learning to label its own training examples using only a small set of domain-independent extraction patterns. KnowItAll was the first published system to carry out extraction from Web pages that was unsupervised, domain-independent, and large-scale. For a given relation, the set of generic patterns was

used to automatically instantiate relation-specific extraction rules, which were then used to learn domain-specific extraction rules. The rules were applied to Web pages identified via search-engine queries, and the resulting extractions were assigned a probability using information-theoretic measures derived from search engine hit counts. Next, KnowItAll used frequency statistics computed by querying search engines to identify which instantiations were most likely to be bona fide members of the class. For instance, KnowItAll was able to confidently label China, France, and India as members of the class Country while correctly knowing that the existence of the sentence, "Garth Brooks is a country singer" did not provide sufficient evidence that "Garth Brooks" is the name of a country. KnowItAll is self-supervised; instead of utilizing hand-tagged training data, the system selects and labels its own training examples and iteratively bootstraps its learning process. KnowItAll is relation-specific in the sense that it requires a laborious bootstrapping process for each relation of interest, and the set of relations has to be named by the human user in advance. This is a significant obstacle to open-ended extraction because unanticipated concepts and relations are often encountered while processing text.

2.1.2 Open IE systems

While most IE work has focused on a small number of relations in specific preselected domains, certain corpora (e.g., encyclopaedias, news stories, email, and the Web itself) are unlikely to be amenable to these methods (Etzioni, Banko et al. 2008). Traditional IE requires pre-specifying a set of relations of interest and then providing training examples for each. Open Information Extraction (Open IE) (Banko and Etzioni 2008) is relation-independent, and instead extracts all relations by learning a set of lexico-syntactic patterns.

The challenge of Web extraction led to the creation of the Open IE field, a novel extraction paradigm that tackles an unbounded number of relations, eschews domain-specific training data, and scales linearly (with low constant factor) to handle Web-scale corpora. For example, an Open IE system might operate in two phases. First, it would learn a general model of how relations are expressed in a particular language. Second, it could utilize this model as the basis of a relation-independent extractor whose sole input is a corpus and whose output is a set of extracted tuples that are instances of a potentially unbounded set of relations. Such an Open IE system would learn a general model of how relations are expressed (in a particular language), based on unlexicalized features such as part-of-speech tags (for example, the identification of a verb in the surrounding context) and domain-independent regular expressions (for example, the presence of capitalization and punctuation). When using the Web as a corpus, the relations of interest are not known prior to extraction, and their number is immense. Thus an Open IE system cannot rely on hand-labelled examples of each relation.

The most representative example of this kind of systems is TextRunner (Banko

and Etzioni 2008; Etzioni, Banko et al. 2008). TextRunner extracts high-quality information from sentences in a scalable and general manner. Instead of requiring relations to be specified in its input, TextRunner learns the relations, classes, and entities from its corpus using its relation-independent extraction model. TextRunner's first phase uses domain-specific examples that have been tagged. With this machine-learning approach, an IE system uses a domain-independent architecture and sentence analyzer. When the examples are fed to machine-learning methods, domain-specific extraction patterns can be automatically learned and used to extract facts from text. Rather than demand hand-tagged corpora, these systems required a user to specify relation-specific knowledge through a small set of seed instances known to satisfy the relation of interest, or a set of hand-constructed extraction patterns to begin the training process. For instance, by specifying the set Bolivia, city, Colombia, district, Nicaragua over a corpus in the terrorism domain, these IE systems learned patterns (for example, headquartered in <x>, to occupy <x>, and shot in <x>) that identified additional names of locations. Nevertheless, the amount of manual effort still scales linearly with the number of relations of interest, and these target relations must be specified in advance.

2.2 Ontologies and Information Extraction

IE's ultimate goal, which is the detection and extraction of relevant information from textual documents, depends on proper understanding of text resources. Rule-based IE systems are limited by the rigidity and ad-hoc nature of the manually composed extraction rules. As a result, they present a very limited semantic background.

The role of semantics in IE is often reduced to very shallow semantic labelling. Semantic analysis is considered more as a way to disambiguate syntactic steps than as a way to build a conceptual interpretation. Today, most of the IE systems that involve semantic analysis exploit the simplest part of the whole spectrum of domain and task knowledge, that is to say, named entities. However, the growing need for IE application to domains such as functional genomics that require more text understanding pushes towards more sophisticated semantic knowledge resources and thus towards ontologies viewed as conceptual models.

In recent years, ontologies have emerged as a new paradigm to model and formalize domain knowledge in a machine readable way. In (Studer, Benjamins et al. 1998) an ontology is defined as "a formal, explicit specification of a shared conceptualization". Conceptualization refers to an abstract model of some phenomenon in the world by having identified its relevant concepts. Explicit means that the type of concepts identified, and the constraints of their use, are explicitly defined. Formal refers to the fact that the ontology should be machine-readable. Shared reflects the notion that an ontology captures consensual knowledge, that is, not a personal view of the target phenomenon of some particular individual, but one

accepted by a group.

Ontologies are designed for being used in applications that need to process the content of information, as well as to reason about it, instead of just presenting information to humans. They permit greater machine interpretability of content than that supported by XML, RDF and RDF Schema (RDF-S), by providing additional vocabulary along with a formal semantics. So, ontologies represent an ideal knowledge background in which to base text understanding and enable the extraction of relevant information. This may enable the development of more flexible and adaptive IE systems than those relying on manually composed extraction rules (both based on linguistic constructions or document structure).

In (Yildiz and Miksch 2007), it is argued that ontologies can assist both manually or semi-automatically constructed rule-based IE systems. On the one hand, the knowledge engineer can commit to the ontology, which would guarantee that the extraction rules are tailored to extract the kind of information represented in the ontology. On the other hand, an annotator can commit to the ontology and annotate only parts of text that are relevant from the ontology's point of view.

Global scale initiatives (e.g. the Semantic Web (Berners-Lee, Hendler et al. 2001)) have brought the development of ontologies for many domains. Nowadays, thousands of domain ontologies are freely available through the Web (Ding, Finin et al. 2004) and big, detailed and consensued general-purpose ontologies (such as WordNet (Fellbaum 1998)) have been developed.

In this section, it is stated how ontologies have been applied in the process of IE from textual documents, specially focusing on domain independent approaches.

2.2.1 Ontology exploitation for IE

IE and ontologies are involved in two main and related tasks (Nedellec and Nazarenko 2005):

- Ontology is used for Information Extraction: IE needs ontologies as part of the understanding process for extracting the relevant information;
- Information Extraction is used for populating and enhancing the ontology: texts are useful sources of knowledge to design and enrich ontologies.

These two tasks, as can be seen in Figure 2, can be combined in a cyclic process: ontologies are used for interpreting the text at the right level for IE and IE extracts new knowledge from text, to be integrated in the ontology.

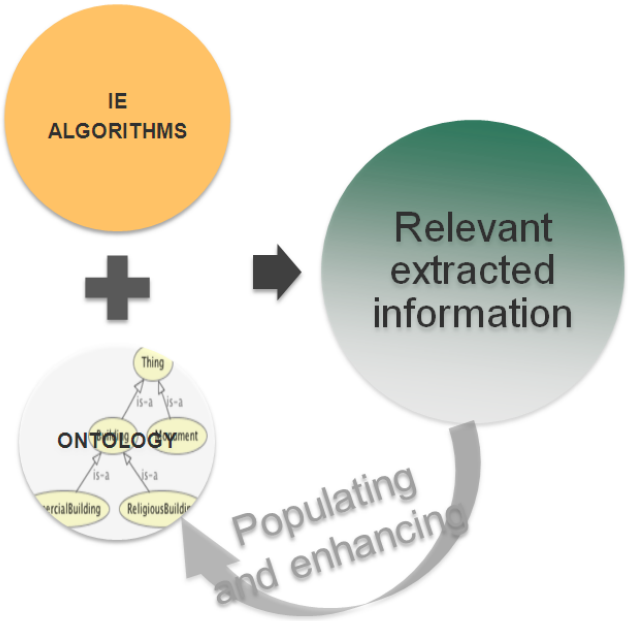


Figure 2 Ontology exploitation for IE (cyclic process)

An ontology identifies the entities that exist in a given domain and specifies their essential properties. It does not describe the spurious properties of these entities. On the contrary, the goal of IE is to extract factual knowledge to instantiate one or several predefined forms. The structure of the form is a matter of the ontology whereas the values of the filled template usually reflect factual knowledge that is not part of the ontology.

Whether one wants to use ontological knowledge to interpret natural language or to exploit written documents to create or update ontologies, in any case, the ontology has to be connected to linguistic phenomena. A large effort has been devoted in traditional IE systems based on local analysis to the definitions of extraction rules that achieve this anchoring. In more powerful IE systems, the ontological knowledge is more explicitly stated in the rules that bridge the gap between the word level and text interpretation. As such, an ontology is not a purely conceptual model, it is a model associated to a domain-specific vocabulary and grammar. In the IE framework, we consider that this vocabulary and grammar are part of the ontology, even when they are embodied in extraction rules.

The complexity of the linguistic anchoring of ontological knowledge is well known. A concept can be expressed by different terms and many words are ambiguous. Rhetoric, such as lexicalized metonymies or elisions, introduces conceptual shortcuts at the linguistic level and must be elicited to be interpreted into

domain knowledge. These phenomena, which illustrate the gap between the linguistic and the ontological levels, strongly affect IE performance. This explains why IE rules are so difficult to design.

IE does not require a whole formal ontological system but parts of it only. The ontological knowledge involved in IE can be viewed as a set of interconnected and concept-centered descriptions, or "conceptual nodes". In conceptual nodes the concept properties and the relations between concepts are explicit. These conceptual nodes should be understood as chunks of a global knowledge model of the domain.

In general, the template or form to be fulfilled by IE is a partial model of world knowledge. IE forms are also classically viewed as a model of a database to be filled by the instances extracted. In (Nedellec and Nazarenko 2005) different levels of ontological knowledge are distinguished:

- The referential domain entities and their variations are listed in "flat ontologies". This is mainly used for entity identification and semantic tagging of character strings in documents.
- At a second level, the conceptual hierarchy improves normalization by enabling more general levels of representation.
- More sophisticated IE systems also make use of chunks of a domain model (i.e. conceptual nodes), in which the properties and interrelations of entities are described. The projection of these relations on the text both improves the NL processes and guides the instantiation of conceptual frames, scenarios or database tuples. The corresponding rules are based either on lexicosyntactic patterns or on more semantic ones.
- The domain model itself is used for inference. It enables different structures to be merged and the implicit information to be brought to light.

In the following paragraphs those elements are discussed in more detail.

Sets of entities

Recognizing and classifying named entities in texts requires knowledge on the domain entities. Specialized lexical or keyword lists are commonly used to identify the referential entities in documents. Three main objectives of these specialized lexicons can be distinguished: semantic tagging, naming normalization and linguistic normalization.

- Semantic tagging. List of entities are used to tag the text entities with the relevant semantic information. In the ontology or lexicon, an entity (e.g. Tony Bridge) is described by its type (the semantic class to which it belongs, here PERSON) and by the list of the various textual forms (typographical variants, abbreviations, synonyms) that may refer to it3 (Mr. Bridge, Tony Bridge, T. Bridge). However, exact character strings are often not reliable enough for a precise entity identification and semantic tagging. Polysemic words belong to different semantic classes. In the above

example, the string “Bridge” could also refer to a bridge named “Tony”. The connection between the ontological and the textual levels must therefore be stronger. Identification and disambiguation contextual rules can be attached to named entities.

- Naming normalization. As a by-effect, these resources are also used for normalization purposes. For instance, the various forms of Mr. Bridge will be tagged as MAN and associated with its canonical name form: Tony Bridge (<PERSON id=Tony Bridge>). This avoids rule overfitting by enabling specific rules to be abstracted.
- Linguistic normalization. Beyond typographical normalization, the semantic tagging of entities contributes to sentence normalization at a linguistic level. It solves some syntactic ambiguities, e.g. if cotA is tagged as a gene, in the sentence “the stimulation of the expression of cotA”, knowing that a gene can be “expressed” helps to understand that “cotA” is the patient of the expression rather than its agent or the agent of the stimulating action. Semantic tagging is also traditionally used for anaphora resolution.

Hierarchies

Beyond lists of entities, ontologies are often described as hierarchies of semantic or word classes. Traditionally, IE focuses on the use of word classes rather than on the use of the hierarchical organization. For instance, in WordNet (Fellbaum 1998), the word classes (synsets) are used for the semantic tagging and disambiguation of words but the hyponymy relation that structures the synsets into a hierarchy of semantic or conceptual classes is seldom exploited for ontological generalization inference. Some ML-based experiments have been done to exploit hierarchies of WordNet and of more specific lexicons, such as UMLS (Freitag 1998). The ML systems learn extraction rules by generalizing from annotated training examples. They relax constraints along two axes, climbing the hyperonym path and dropping conditions. In this way, the difficult choice of the correct level in the hierarchy is left to the systems.

Conceptual nodes

The ontological knowledge is not always explicitly stated as it is in (Gaizauskas and Wilks 1998), which represents an ontology as a hierarchy of concepts, each concept being associated with an attribute-value structure, or in (Embley, Campbell et al. 1998), which describes an ontology as a database relational schema. However, ontological knowledge is reflected by the target form that IE must fill and which represents the conceptual nodes to be instantiated. Extraction rules ensure the mapping between a conceptual node and the potentially various linguistic phrasings expressing the relevant elements of information.

The main difficulty arises from the complexity of the text representation once enriched by the multiple linguistic and conceptual levels. The more expressive the

representation, the larger is the search space for the IE rule and the more difficult the learning. The extreme alternative consists in either selecting the potentially relevant features before learning, with the risk of excluding the solution from the search space, or leaving the system the entire choice, provided that there are enough representative and annotated data to find the relevant regularities. For instance, the former consists in normalizing by replacing names by category labels whereas the latter consists in tagging without removing the names. The learning complexity can even be increased when the conceptual or semantic classes are learned together with the conceptual node information (Yangarber, Grishman et al. 2000).

2.2.2 Ontology-based Information Extraction

We consider ontology-based IE systems as those approaches relying on predefined ontologies in one or several stages of the extraction process. Those approaches are document driven: they start from a particular document (or set of documents) and they try to identify entities found in that context, trying to annotate them according to the input ontology. So, on the contrary to plain IE systems, ontology-based ones are able to specify their output in terms of a pre-existing formal ontology. These systems almost always use a domain-specific ontology in their operation, but we consider a system to be domain-independent if it can operate without modification on ontologies covering a wide range of domains.

So, the problem is very similar to semantic annotation. Annotations represent a specific sort of metadata that provides references between entities appearing in resources and domain concepts modelled in an ontology. Semantic annotation is one fundamental pillar of the Semantic Web (Berners-Lee, Hendler et al. 2001) making it possible for Web-based tools to understand and satisfy the requests of people and machines to exploit Web content.

In this section we refer to both semantic annotation and ontology-based IE indistinctly.

In the last years, several attempts have been made to address the annotation of textual Web content. From the manual point-of-view, several tools have been developed to assist the user in the annotation process such as Annotea (Koivunen 2005), CREAM (Handschuh, Staab et al. 2003), NOMOS (Niekrasz and Gruenstein 2006) or Vannotea (Schroeter, Hunterd et al. 2003). Those systems rely on the skills and will of a community of users to detect and tag entities within Web content. Considering that there are 1 trillion of unique Web pages on the Web (see The Official Google Blog, http://googleblog.blogspot.com/2008/07/we-knew-Web-was-big.html, last access on March 30th, 2010) , it is easy to envisage the unfeasibility of manual annotation of Web resources.

Recently, some authors have focused on addressing the annotation problem by automating some of its stages. As a result, some tools such as Melita (Ciravegna, Dingli et al. 2002) have been developed. It is based on user-defined rules and

previous annotations to suggest new annotations in text. Manually constructed rules are used also in other basic approaches to extract known patterns for annotations (Baumgartner, Flesca et al. 2001). Another preliminary work proposing semi-automating the annotation of Web resources is the work described in (Kiyavitskaya, Zeni et al. 2005). The authors propose the combination of patterns (e.g., addressed to extract objects such as email addresses, phone numbers, dates and prices) to tag the candidates to annotate, and then, this set is annotated by means of a domain conceptual model. That model represents the information of a particular domain through concepts, relationships and attributes (in an entity-relation based syntax). Supervised systems also use extraction rules obtained from a set of pre-tagged data (Califf and Mooney 2003; Roberts, Gaizauskas et al. 2007). WebKB (Cafarella, Downey et al. 2005) and Armadillo (Alfonseca and Manandhar 2002) use supervised techniques to extract information from computer science websites. Likewise, S-CREAM (Cunningham, Maynard et al. 2002) uses machine learning techniques to annotate a particular document with respect to its ontology, given a set of annotated examples.

Supervised attempts are certainly difficult to apply due to the bottleneck introduced by the interaction of a domain expert and the great effort required to compile a large and representative training set.

SmartWeb (Buitelaar, Cimiano et al. 2008) resolves the issue of not having pre-existing mark-up to learn from by using class and subclass names from a previously defined ontology. Those are used as examples to learn contexts. In this way, instances can be identified, as they present similar contexts.

Complete automatic and unsupervised systems are rare. SemTag (Dill, Eiron et al. 2003) performs automated semantic tagging from large corpora based on the Seeker platform for text analysis and tagging large number of pages with the terms included in a domain ontology named TAP. This ontology contains lexical and taxonomic information about music, movies, sports, health, and other issues, and SemTag detects the occurrence of these entities in Web pages. It disambiguates using neighbour tokens and corpus statistics, picking the best label for a token. KIM (Kiryakov, Popov et al. 2004) is another example of unsupervised domain-independent system. It scans documents looking for entities corresponding to instances in its input ontology.

Another interesting annotation application is presented in (Michelson and Knoblock 2007). In this case, the authors use a reference set of elements (e.g., online collections containing structured data about cars, comics or general facts) to annotate ungrammatical sources like texts contained in posts. First of all, the elements of those posts are evaluated using the TF-IDF metric. Then, the most promising tokens are matched with the reference set. In both cases, limitations may be introduced by the availability and coverage of the background knowledge (i.e., ontology or reference sets). From the applicability point-of-view, Pankow (Cimiano, Handschuh et al. 2004) is the most promising system. It uses a range of well-studied

syntactic patterns to mark-up candidate phrases in Web pages without having to manually produce an initial set of marked-up Web pages, and without depending on previous knowledge. The context driven version, C-Pankow (Cimiano, Ladwig et al. 2005), improves the first by reducing the number of queries to the search engine. However, the final association between text entities and a possible domain ontology is not addressed.

There exist other systems which present a more ad-hoc design and are focused on a specific domain of knowledge, exploiting predefined and expected corpus structures, rules and domain knowledge. In (Maedche, Neumann et al. 2003) an IE system focused on the Tourism domain is proposed. They combine lexical knowledge, extraction rules and ontologies in order to extract information in the form of instantiated concepts and attributes that are stored in an ontology-like fashion (e.g. hotel names, number of rooms, prices, etc.). The most interesting feature is the fact that the pre-defined knowledge structures are extended as a result of the IE extraction process allowing to improve and complete them. They use several ontology learning techniques already developed for the OntoEdit system (Staab and Maedche 2000). The process starts with a shallow IE model given as baseline. Then, a domain specific corpus is selected. The corpus is processed with the core IE system. Based on this data, one is able to use different learning approaches in a semi-supervised fashion embedded into the Ontology learning framework. As a result, the process is extended. The human expert has to validate each extension before continuing.

Feilmayr et. al. (Feilmayr, Parzer et al. 2009) propose an ontology-based IE system. They analyse the heterogeneities of individually maintained accommodation websites and discuss the IE techniques in the Tourism domain. As a result, they present a rule/ontology-based IE approach able to cope with the given heterogeneities. A domain-dependent crawler collects Web pages corresponding to accommodation websites. This corpus is passed to an extraction component based on the GATE framework (Cunningham, Maynard et al. 2002) which provides a number of text engineering components. It performs an annotation of Web pages in the corpus, supported by a domain-dependent ontology and rules. Extracted tokens are ranked as a function of their frequency and relevancy for the domain.

Another domain-dependent system is SOBA (Buitelaar, Cimiano et al. 2008), a sub-component of the SmartWeb (a multi-modal dialog system that derives answers from unstructured resources such as the Web), which automatically populates a knowledge base with information extracted from soccer match reports found on the Web. The extracted information is defined with respect to an underlying ontology. The SOBA system consists of a Web crawler, linguistic annotation components and a module for the transformation of linguistic annotation into an ontology-based representation. The first component enables the automatic creation of a soccer corpus, which is kept up-to-date on a daily basis. Text, images and semi-structured data are compiled. Linguistic annotation is based in finite-state techniques and unification-based algorithms. It implements basic grammars for the annotation of

persons, locations, numerals and date and time expressions. On the top, rules for extraction of soccer-specific entities, such as actors in soccer, teams and tournaments are implemented. Finally, data is transformed into ontological facts, by means of tabular processing (wrapper-like techniques are applied) and text matching (by means of F-logic structures specified in a declarative form).

(Li and Ramani 2007) proposes to use shallow natural language processing and domain-specific ontologies (applied to the manufacturing and vehicle domains) to automatically construct a structured representation from a set of unstructured documents. Concepts and relations are identified in the text by means of linguistic patterns. The result is stored in an ontology-like fashion. Apart from the basic linguistic analysis of text (tokenization, POS tagging and chunking), which results in the extraction of noun and verb phrases, the system maps them to the input ontology by simple word matching. Breadth first search is used to search for concepts in the domain ontology which match the extracted entities. Extracted noun phrases are compared against all the concepts in the domain ontology, whereas verb phrases are matched against a manufacturing taxonomy. In the case of multiple matchings, the one with the highest amount of matchings in the same sentence is selected.

The basic idea of the approach by Yildiz and Miksch (Yildiz and Miksch 2007) is to use the information on the input ontology to construct automatically a set of extraction rules to be used by the information extraction system. They look on the text for the words that appear in the name of the concepts, the name of the properties and the comment section of the concepts and attributes. For each appearance of one of these words, they apply rules (regular expressions related to the datatype of each property, as specified in the ontology) to the word's neighbourhood to find appropriate values. For instance, if there is an ontology on digital cameras in which the Digital Camera class has an Optical Zoom property (of the float type), the system looks for the string "optical zoom" in the text and searches for a float numerical value near it.

2.2.3 Ontology-driven Information Extraction

The methods described in the previous section may be qualified as document-driven, since they analyse sequentially a given set of documents available in a corpus, trying to annotate the information of those documents with respect to the input ontology. A complementary approach, which can be qualified as ontology-driven, is commented in this section. The basic idea of the techniques in this category is to focus the processing on the ontology basic elements (classes, relations), leveraging this knowledge to find resources that can be analysed to obtain useful information (in most cases, instances of the ontology classes). As commented in (McDowell and Cafarella 2008), this kind of methods presents some benefits:

- Focusing on the ontology components seems a natural way to exploit all kinds of ontological data (e.g. using synonyms to broaden the search for

documents to be analysed).

- These systems can consider a huge amount of different resources (e.g. the Web), and are not constrained by a limited corpus of documents.
- The systems concentrate all their resources on searching directly for information related to the ontology components, rather than having to analyse a potentially large number of documents that do not contain interesting information.

One of the most well-known examples of ontology-driven information extraction systems in OntoSyphon (McDowell and Cafarella 2008), a domain-independent and unsupervised system which focuses on finding instances of the classes of the input ontology. For each class of the ontology, the following steps are taken:

- Use a basic set of Hearst patterns (Hearst 1992) to generate lexico-syntactic phrases that permit to obtain candidates to instances of the class. For example, for the Bird class, the patterns used would be "birds such as …", "birds including …", "birds especially …", "… and other birds", "… or other birds".
- Use those phrases in a Web search engine (or in a simplified setting such as the Binding Engine (Cafarella, Downey et al. 2005)) to extract the candidate instances.
- Evaluate those candidates to assess which of them have a good chance of being instances of the class. The evaluation measures proposed in (McDowell and Cafarella 2008) depend basically on the number of patterns from which a given candidate has been obtained and the number of hits of each candidate (redundancy is taken as a signal that the candidate is probably good), although more complex evaluations based on the urn model and on variations of PMI (Turney 2001) are also proposed.

The work on information extraction by Vicient (Vicient 2009) is also guided by the classes of an input ontology, although the set of Web pages to be analysed is fixed and no Web searches are performed. His methodology is domain-independent, but the work centred the analysis in a Tourism ontology, which was manually constructed. The aim of this work, very much related to the objectives of the DAMASK project, was to generate a matrix in which each row corresponded to a destination city, each column was related to a class of the ontology, and each cell of the matrix showed the subclasses of the class on the column which denote elements that are present in the city on the row. For instance, if the row is London and the column is Religious-Building, the related cell would contain a list such as "Cathedral, Mosque, Synagogue, Abbey, Church", which are subclasses of Religious Building that are represented by real buildings in London. For each class of the ones considered in the matrix columns (selected by the user from the input ontology), the systems analyzes the Wikipedia pages related to the touristic destinations in the following way:

- All the subclasses of the class are recursively searched in the basic text of the page (e.g. “St.Paul’s Cathedral” identifies an item of the Cathedral class, and “London Central Mosque” an instance of the Mosque class).
- The subclasses are also searched in the list of categories associated to the Wikipedia page.
- The text associated to each of the images of the page is also compared with the subclasses of the class.

The numerical attributes related to the CityClass are instantiated by analyzing the infoBox that appears at the beginning of the Wikipedia page. Although the work may be considered as a first step in the direction of the DAMASK objectives, it has to be noticed that the identification of the subclasses of each class within the Web pages is purely syntactical.

Van Hague et al. (van Hage, Katrenko et al. 2005) present an ontology-driven domain-independent method that, although it is not focused precisely on Information Extraction but rather on Ontology Mapping, uses similar ideas. Their aim is to find a mapping between pairs of concepts belonging to two input ontologies. For each pair (C1, C2), where C1 is a class of the first ontology and C2 is a class of the second ontology, they perform the following tasks:

- Use a basic set of hyponymy-detector Hearst patterns (C1 such as C2, such C1 as C2, C1 including C2, etc).
- Send the patterns to a Web search engine, and collect the hit counts obtained in each case.
- Accept all hyponymy relations supported by a number of hits above a certain threshold.

Another approach for ontology-driven information extraction is given in (Geleijnse, Korst et al. 2006). In this work the aim is to find instances of the classes of the input ontology. The procedure follows these steps:

- Select one of the binary relations of the ontology and one instance corresponding to the domain or the range of the relation (for example, the relation “acts in” –between Actors and Movies- and an instance of Actor, “Sean Connery”).
- The system contains a set of manually-constructed text patterns associated to the relation (in the same example, the relation “acts in” is associated to the pattern “[Movie] starring [Actor], [Actor] and [Actor]”). Take each pattern and apply it to the instance (e.g. “[Movie] starring Sean Connery, [Actor] and [Actor]”).
- Send each of these instantiated patterns to a Web search engine, and collect candidates to instances of the classes appearing in the pattern (in the example, with the previous pattern we would obtain candidates to instances of the classes Movie and Actor).

- Check the correctness of each candidate, by sending to the Web search engine phrases expressing the instance-class relation (which are constructed semi-automatically) and accepting the instance candidate when the number of hits obtained exceeds a certain threshold.

A similar approach to ontology-driven population is reported by Matuszek et al. (Matuszek, Witbrock et al. 2005). This work is framed in the Cyc project, the ambitious effort that has been going on for some decades to formalize all the world's commonsense knowledge. In particular, the authors have developed techniques for automatically finding instances of the components (domain, range) of the relations on the ontology. Their approach follows these steps:

- Choose a query that represents information that wants to be found out (e.g. the Prime Minister of a certain country). The authors have limited the search to 134 binary predicates.
- Translate the query into a search string. The system contains 233 manually created generation templates for the 134 chosen predicates.
- Send the query to a Web search engine, and detect the class instance candidates.
- A candidate is deemed as correct if it successfully passes three tests: it does not create any logical inconsistency with the knowledge already present in Cyc, a specifically generated search string containing the candidate and the class provides enough hits, and a human curator finally validates the candidate.

The main drawback of the last two methods is that they contain some steps that cannot be made automatically, and therefore they require a certain amount of manual work before they can be executed for a given domain ontology.

2.3 Summary

Information Extraction (IE) methods aim to find specific items of information within electronic resources (usually text documents), by applying some kind of extraction rules. These rules may be given by a domain expert, may be learnt from documents tagged by a domain expert, or may be learnt directly from the texts through the use of some generic information patterns. In the DAMASK project we are interested in this last option, as we want to develop an unsupervised IE framework.

The relation between ontologies and IE is twofold: on the one hand, the semantic knowledge given by a domain ontology may guide the IE process (as in the case of the DAMASK project) and, on the other hand, the IE results may help to improve or enrich an initial domain ontology.

In this document we have considered two different kinds of methods involving

ontologies and IE. In the ontology-based (or document-driven) methods, each document of the corpus is analysed sequentially, and the aim is to annotate each document by relating specific pieces of information to the concepts, instances and relations in the ontology. On the contrary, in the ontology-driven techniques the idea is to consider each of the ontological elements and to use them to search for resources (e.g. Web pages) that can provide interesting information related to each component of the ontology. Some initial work developed in our group (Vicient 2009) along the initial steps of the DAMASK project felt into this category.

3 Learning techniques, tools and work environment

This section introduces the main techniques, tools and concepts needed for a correct understanding of the implemented solution, that are applied in this Master thesis and it presents the work environment justifying the main reasons for using the Web as a corpus.

This chapter is structured as follows:

- §3.1 presents the work environment. First, it is argued that the Web can be a valid knowledge learning repository thanks to the huge amount of information available for every possible domain and its high redundancy. Moreover, this redundancy may allow lightweight analytic approaches to obtain good quality results maintaining scalability and efficiency in this enormous and noisy environment (Paşca 2005). Moreover, Web snippets are explained and its use is argued to achieve the goals of the work with a lightweight analysis. Finally, the main characteristics of Wikipedia, which will be used as an example of semi-structured resource, are exposed.
- In §3.2, different useful techniques used on this work are commented. First it introduces Natural Language as an Artificial Intelligence research area and exposes the main techniques and tools for exploiting it. Then, the main works using lexico-syntactic patterns and in which type of ontology concepts they can be applied are stated. Finally, several heuristics for exploiting the statistics provided by Web search engines are presented.
- §3.3 introduces the paradigm of ontologies and presents WordNet as a useful tool to extract related terms of any concept.

3.1 Work environment

This section presents the work environment. It is divided in three parts: the Web as a corpus, Web snippets and Wikipedia.

3.1.1 The Web as a corpus

Many classical knowledge acquisition techniques present performance limitations due to the typically reduced corpus used (Brill 2003). This idea is supported by current social studies as (Surowiecki 2004), in which it is argued that collective knowledge is much more powerful than individual knowledge. The Web is the biggest repository of information available (Brill 2003). This fact can represent a great deal when using it as a corpus for knowledge acquisition.

Apart from the huge amount of information available, another feature that characterizes the Web is its high redundancy. This fact has been mentioned by several authors and it is especially important because the amount of repetition of information can represent a measure of its relevance (Brill 2003; Ciravegna, Dingli et al. 2003; Etzioni, Cafarella et al. 2004; Rosso, Montes et al. 2005). This can be a good approach to tackle the problem of untrustworthiness of the resources: we cannot trust the information contained in an individual website, but we can give more confidence to a fact that is enounced by a considerable amount of possibly independent sources. This fact is also related to the consensus that the extracted knowledge should present: implicit consensus can be achieved as concepts are selected among the terms that are frequently employed in documents produced by the virtual community of users (Navigli and Velardi 2004).

Thanks to those characteristics, the Web has demonstrated its validity as a corpus for research (Volk 2002; Jarmasz and Szpakowicz 2003) with successful results in many areas: question answering (Brill, Lin et al. 2001; Kwok, Etzioni et al. 2001), question classification (Solorio, P\ et al. 2004), anaphora resolution (Bunescu ; Markert, Modjeska et al. 2003) , Prepositional Phrase treatment(Volk 2001; Calvo and Gelbukh 2003), and ontology enrichment (Agirre, Ansa et al. 2000).

3.1.2 Web snippets

Web Snippets are fragments of text returned when querying a Web search engine. They are used to obtain previews of the information contained in the Web. Those are presented in the form of the context in which the queried keyword(s) is(are) presented (see Figure 3). These previews, typically called snippets, even offering a narrow context, are informative enough to extract related knowledge without accessing the Web's content.

Figure 3 Snippet of a website obtained by Google for the Tarragona domain.

In this work, snippets can be particularly useful either for pattern-based extraction of candidates (only considering a short context for the constructed query) or for the semantic disambiguation of terms to extract synonyms using WordNet(see section 4.1.3.2.2).

3.1.3 Wikipedia

Wikipedia is a free, Web-based, collaborative, multilingual encyclopaedia project supported by the non-profit Wikimedia Foundation. Its 18 million articles (over 3.6 million in English) have been written collaboratively by volunteers around the world, and almost all of its articles can be edited by anyone with access to the site.

In this work, the proposed methodology is able to extract information from different kinds of Web resources (i.e., plain texts and semi-structured documents). To exemplify the extraction from semi-structured resources, Wikipedia has been used. Concretely, Wikipedia is useful to evaluate the proposed methodology due to its properties. Wikipedia is a semi-structured Web resource and brings metadata to its contents. Metadata are a set of descriptive elements which are used to identify documents or digital resources. For some areas in computer science such Information Extraction, Information Retrieval and the Semantic Web, metadata are labels which gives semantics to the contents that are being annotated.

Moreover, the Wikipedia is particularly useful because of its link structure. Wikipedia links brings information about relations and they connect the textual contents with conceptual levels. There exist two different types of links which deserve to be mentioned: internal links and category links.

On one hand, internal links (also known as pagelinks or Wikilinks) represent links to other Wikipedia articles. This fact means that, in a Wikipedia article, the main features or facts about the real entity which the article is talking about are linked with other Wikipedia articles. The advantages of this characteristic are that these relations give a kind of implicit information (i.e. two articles are related) and that users can navigate among all related articles in an easy way.

On the other hand, category links are used to organize the knowledge contained in Wikipedia by grouping together pages on similar subjects. Categories are meant to be a navigational system that helps readers quickly move from one related article to another within a related subject area. Wikipedia's category system can be thought

of as consisting of overlapping trees. Any category may branch into subcategories, and it is possible for a category to be a subcategory of more than one parent. (A is said to be a parent category of B when B is a subcategory of A). Mathematically speaking, this means that the system approximates a directed acyclic graph.

For example, the Wiki about "Barcelona" has an internal link to "The Sagrada Familia" article which is categorized as Antoni Gaudí buildings, Buildings and structures under construction, Churches in Barcelona, Visitor attractions in Barcelona, World Heritage Sites in Spain, Basilica churches in Spain, etc. The conclusion is that Barcelona is related with Sagrada Familia and this last one can be categorized as a church or basilica (similar concepts), as a building (concept which is in an higher level than church and basilica but is directly related with those concepts by a taxonomic relationship) and as a visitor attraction or World Heritage Site (concepts that are not related with the other ones).

3.2 Techniques

Following the main implemented techniques in this work are presented. First, the main techniques in Natural Language Parsing are presented in section 3.2.1. In section 3.2.2, Hearst patterns and their applicability are discussed. Finally, the use of statistical measures is stated (section 3.2.3).

3.2.1 Natural Language processing

In the philosophy of language, a natural language (or ordinary language) is any language which arises in an unpremeditated fashion as the result of the innate facility for language possessed by the human intellect. A natural language is typically used for communication, and may be spoken, signed, or written.

Natural language processing (referred as NLP) is the study of mathematical and computational modelling of various aspects of language and the development of a wide range of systems. Research in NLP is highly interdisciplinary, involving concepts in computer science, linguistics, logic, and psychology. NLP has a special role in computer science, particularly in the sub-field of Artificial Intelligence, because many aspects of the field deal with linguistic features of computation and NLP seeks to model language computationally. By applying NLP it is possible to analyse sentences syntactically.

Concerning the analysis of text itself, this work only considers English written resources and exploits some peculiarities of that language to extract knowledge. Therefore, a set of tools and algorithms for analysing English natural language is used for that purpose. Concretely:

- Natural Language Processing Parser: it is the responsible for detecting sentences, tokens and parts of speech (Text processing) and perform the

syntactic analysis or Part-Of-Speech tagging. On one hand, the first component is able to chunk a text in order to find its minimal parts. Once the text is chunked, the different minimal pieces obtained are tagged with a Part-Of-Speech (POS) tagger. On the other hand, Syntactic analyser or Part-Of-Speech tagging allows performing basic morphological and syntactical analyses of particular pieces of text that can contain valuable information. This will provide a way to interpret and extract potentially interesting concepts and relationships. Even though their precision is not perfect and, in consequence, some useful information may be omitted, this is not an important problem thanks to the high redundancy of information in the Web.

- Stemming algorithm: allows obtaining the morphological root of a word for the English language. It is fundamental to avoid the redundancy of extracting the different equivalent morphological forms in which a word can be presented. Some examples of this algorithm can be found in(Rijsbergen, Robertson et al. 1980).
- Stop words analysis: finite list of domain independent words with very general meaning that can be omitted during the analysis. Determinants, prepositions or adverbs are typically contained in this category.

Following subsections explain in detail these points.

3.2.1.1 Natural Language Processing parser

The first step of natural language parsing is to detect sentences from a text. Most natural language processing tools use as default the point (.) as a delimiter to separate sentences. Other delimiters can be used such as comma (,), question mark (?), exclamation (!) and others. Once the sentence detector splits the whole text, the tokenization is the next step. It is in charge of separating the words for the analysis. For example, “don't” should be separated to “do” and “not” for a good further text analysis. After that, the sentence analysis can be applied. Once the tokenization is successfully done, the part of speech tagging or word category disambiguation process is executed. POS tagging is the process of marking up the words in a text (corpus) as corresponding to a particular part of speech such as nouns, verbs, adjectives, etc. The POS tagging is very useful because provides syntactic information of every word in the sentence structure, but it could also be useful in itself when looking for units of meaning in a sentence. The last step is text chunking which consists of dividing a text in syntactically correlated parts of words, like noun groups, verb groups, but does not specify their internal structure, nor their role in the main sentence.

Following, to clarify how sentence analysis works after sentence detection, the sentence “The red book is a black novel” will be analysed. Figure 4 shows the performed analysis. First, the words are marked as corresponding to a particular part

of speech, by means of POS tagging, such as nouns, verbs, adjectives, etc. In this example the tagged components are:

- DT: Determiner
- JJ: Adjective
- NN: Common noun
- VBZ: Verb, 3rd person singular present

After POS tagging, chunking is applied in order to divide the text in syntactically correlated parts of words. In this case only in noun and verb phrases:

- NP: Noun Phrase
- VP: Verb Phrase

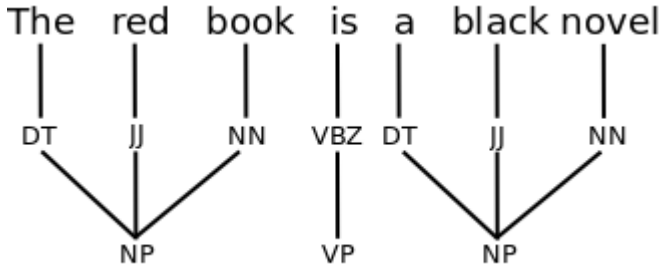


Figure 4 Sentence analysis

In this work, OpenNLP[1] has been used as Natural Language Processing Parser. The text processing tool OpenNLP is a mature Java package that hosts a variety of Natural Language Processing tools which perform sentence detection, tokenization, pos-tagging, chunking and parsing, allowing morphological and syntactical analysis of texts. It is based on maximum entropy models and, in consequence it requires annotation samples. Models of annotation for each task exhaustively trained for the English Language are used (provided "officially" by the developers of the library). It has been used to analyse interesting pieces of Web content (i.e. a pattern matching found within a particular website). Even though the computational cost of this analysis can be high when evaluating large texts, only the particular sentence in which the keyword has been found is considered.

3.2.1.2 Stemming analysis

The automatic removal of suffixes (or stemming) from words in English is of particular interest in the field of information retrieval. The aim of this technique is to find the morphological root of a word.

[1] http://incubator.apache.org/opennlp/

Several algorithms, such as Lemmatisation algorithm, stochastic algorithm, N-gram algorithm or Porter algorithm have been proposed in this research field. The last one, first introduced in (Porter 1997), is one of the most common algorithms applied in information extraction and will be used in this work because of its simplicity.

This technique has been extensively used in order to detect equivalent forms of expressing the same ontological concept for example avoiding duplicity of information to analyse by discarding plurals.

Words	Stemmed word
Connect **Connected** **Connecting** **Connection** **Connections**	Connect
Student **Students**	Student
play **playing**	plai
Child	Child
Children	Children

Table 2 Results of Porter stemming algorithm

Table 2 shows the results of stemming the set of words of column 1, where, the first three sets of words are correctly stemmed and the last one is erroneously considered as two different words and consequently their roots are different. This problem is derived from the fact that Porter algorithm is based on English grammatical rules and the English word exceptions are not taken into account.

3.2.1.3 Stop words

Table 3 shows a list of stop words.

Stop words list
“a”, “about”, “above”, “according”, “across”, “actually”, “ad”, “adj”, “ae”, “af”, “after”, “afterwards”, “ag”, “again”, “against”, “ai”, “al”, “all”, “almost”, “alone”, “along”, “already”, “also”, “although”, “always”, “am”, “among”, “amongst”, “an”, “and”, “another”, “any”, “anyhow”, “anyone”, “anything”, “anywhere”, “ao”, “aq”, “ar”, “are”, “aren”, “aren't”, “around”, “arpa”, “as”, “at”, “au”, “aw”, “az”, “b”, “ba”, “bb”, “bd”, “be”, “became”, “because”, “become”, “becomes”, “becoming”, “been”, “before”, “beforehand”, “begin”, “beginning”, “behind”, “being”, “below”, “beside”, “besides”, “between”, “beyond”, “bf”, “bg”, “bh”, “bi”, “billion”, “bj”, “bm”, “bn”, “bo”, “both”, “br”, “bs”, “bt”, “but”, “buy”, “bv”, “bw”, “by”, “bz”, “c”, “ca”, “can”, “can't”, “cannot”, “caption”, “cc”, “cd”, “cf”, “cg”, “ch”, “ci”, “ck”, “cl”, “click”, “cm”, “cn”, “co”, “co.”, “com”, “copy”, “could”, “couldn”, “couldn't”, “cr”, “cs”, “cu”, “cv”, “cx”, “cy”, “cz”, “d”, “de”, “did”, “didn”, “didn't”, “dj”, “dk”, “dm”, “do”, “does”, “doesn”, “doesn't”, “don”, “don't”, “down”, “during”, “dz”, “e”, “each”, “ec”, “edu”, “ee”, “eg”, “eh”, “eight”, “eighty”, “either”, “else”, “elsewhere”, “end”, “ending”, “enough”, “er”, “es”, “et”, “etc”, “even”, “ever”, “every”, “everyone”, “everything”, “everywhere”, “except”, “f”, “few”, “fi”, “fifty”, “find”, “first”, “five”, “fj”, “fk”, “fm”, “fo”, “for”, “former”, “formerly”, “forty”, “found”, “four”, “fr”, “free”, “from”, “further”, “fx”, “g”, “ga”, “gb”, “gd”, “ge”, “get”, “gf”, “gg”, “gh”, “gi”, “gl”, “gm”, “gmt”, “gn”, “go”, “gov”, “gp”, “gq”, “gr”, “gs”, “gt”, “gu”, “gw”, “gy”, “h”, “had”, “has”, “hasn”, “hasn't”, “have”, “haven”, “haven't”, “he”, “he'd”, “he'll”, “he's”, “help”, “hence”, “her”, “here”, “here's”, “hereafter”, “hereby”, “herein”, “hereupon”, “hers”, “herself”, “him”, “himself”, “his”, “hk”, “hm”, “hn”, “home”, “homepage”, “how”, “however”, “hr”, “ht”, “htm”, “html”, “http”, “hu”, “hundred”, “i”, “i'd”, “i'll”, “i'm”, “i've”, “i.e.”, “id”, “ie”, “if”, “il”, “im”, “in”, “inc”, “inc.”, “indeed”, “information”, “instead”, “int”, “into”, “io”, “iq”, “ir”, “is”, “isn”, “isn't”, “it”, “it's”, “its”, “itself”, “j”, “je”, “jm”, “jo”, “join”, “jp”, “k”, “ke”, “kg”, “kh”, “ki”, “km”, “kn”, “kp”, “kr”, “kw”, “ky”, “kz”, “l”, “la”, “last”, “later”, “latter”, “lb”, “lc”, “least”, “less”, “let”, “let's”, “li”, “like”, “likely”, “lk”, “ll”, “lr”, “ls”, “lt”, “ltd”, “lu”, “lv”, “ly”, “m”, “ma”, “made”, “make”, “makes”, “many”, “maybe”, “mc”, “md”, “me”, “meantime”, “meanwhile”, “mg”, “mh”, “microsoft”, “might”, “mil”, “million”, “miss”, “mk”, “ml”, “mm”, “mn”, “mo”, “more”, “moreover”, “most”, “mostly”, “mp”, “mq”, “mr”, “mrs”, “ms”, “msie”, “mt”, “mu”, “much”, “must”, “mv”, “mw”, “mx”, “my”, “myself”, “mz”, “n”, “na”, “namely”, “nc”, “ne”, “neither”, “net”, “netscape”, “never”, “nevertheless”, “new”, “next”, “nf”, “ng”, “ni”, “nine”, “ninety”, “nl”, “no”, “nobody”, “none”, “nonetheless”, “noone”, “nor”, “not”, “nothing”, “now”, “nowhere”, “np”, “nr”, “nu”, “nz”, “o”, “of”, “off”, “often”, “om”, “on”, “once”, “one”, “one's”, “only”, “onto”, “or”, “org”, “other”, “others”, “otherwise”, “our”, “ours”, “ourselves”, “out”, “over”, “overall”, “own”, “p”, “pa”, “page”, “pe”, “per”, “perhaps”, “pf”, “pg”, “ph”, “pk”, “pl”, “pm”, “pn”, “pr”, “pt”, “pw”, “py”, “q”, “qa”, “r”, “rather”, “re”, “recent”, “recently”, “reserved”, “ring”, “ro”, “ru”, “rw”, “s”, “sa”, “same”, “sb”, “sc”, “sd”, “se”, “seem”, “seemed”, “seeming”, “seems”, “seven”, “seventy”, “several”, “sg”, “sh”, “she”, “she'd”, “she'll”, “she's”, “should”, “shouldn”, “shouldn't”, “si”, “since”, “site”, “six”, “sixty”, “sj”, “sk”, “sl”, “sm”, “sn”, “so”, “some”, “somehow”, “someone”, “something”, “sometime”, “sometimes”, “somewhere”, “sr”, “st”, “still”, “stop”, “su”, “such”, “sv”, “sy”, “sz”, “t”, “taking”, “tc”, “td”, “ten”, “text”, “tf”, “tg”, “test”, “th”, “than”, “that”, “that'll”, “that's”, “the”, “their”, “them”, “themselves”, “then”, “thence”, “there”, “there'll”, “there's”, “thereafter”, “thereby”, “therefore”, “therein”, “thereupon”, “these”, “they”, “they'd”, “they'll”, “they're”, “they've”, “thirty”, “this”, “those”, “though”, “thousand”, “three”, “through”, “throughout”, “thru”, “thus”, “tj”, “tk”, “tm”, “tn”, “to”, “together”, “too”, “toward”, “towards”, “tp”, “tr”, “trillion”, “tt”, “tv”, “tw”, “twenty”, “two”, “tz”, “u”, “ua”, “ug”, “uk”, “um”, “under”, “unless”, “unlike”, “unlikely”, “until”, “up”, “upon”, “us”, “use”, “used”, “using”, “uy”, “uz”, “v”, “va”, “vc”, “ve”, “very”, “vg”, “vi”, “via”, “vn”, “vu”, “w”, “was”, “wasn”, “wasn't”, “we”, “we'd”, “we'll”, “we're”, “we've”, “web”, “webpage”, “website”, “welcome”, “well”, “were”, “weren”, “weren't”, “wf”, “what”, “what'll”, “what's”, “whatever”, “when”, “whence”, “whenever”, “where”, “whereafter”, “whereas”, “whereby”, “wherein”, “whereupon”, “wherever”, “whether”, “which”, “while”, “whither”, “who”, “who'd”, “who'll”, “who's”, “whoever”, “whole”, “whom”, “whomever”, “whose”, “why”, “will”, “with”, “within”, “without”, “won”, “won't”, “would”, “wouldn”, “wouldn't”, “ws”, “www”, “x”, “y”, “ye”, “yes”, “yet”, “you”, “you'd”, “you'll”, “you're”, “you've”, “your”, “yours”, “yourself”, “yourselves”, “yt”, “yu”, “z”, “za”, “zm”, “zr”, “z”, “hoc”, “ad”

Table 3 Stop words list

3.2.2 Linguistic patterns

Instance-Concept relations refers to “is-a” relationships. There exist many approaches for performing the task of detecting this kind of relations. However, as

this work is focused on an unsupervised, domain independent approach, appropriate techniques should be employed. As stated in (Cimiano, Handschuh et al. 2004), three different learning paradigms can be exploited. First, some approaches rely on the document-based notion of term subsumption (Sanderson and Croft 1999). Secondly, some researchers claim that words or terms are semantically similar to the extent to which they share similar syntactic contexts (Caraballo 1999; Bisson, N\'dellec et al. 2000). Finally, several researchers have attempted to find taxonomic relations expressed in texts by matching certain patterns associated to the language in which documents are presented(Berland and Charniak 1999; Ahmad, Tariq et al. 2003).

Pattern-based approaches are heuristic methods using regular expressions that have been successfully applied in information extraction. The text is scanned for instances of distinguished lexical-syntactic patterns that indicate a relation of interest. This is especially useful for detecting specialisations of concepts that can represent is-a (taxonomic) relations (Hearst 1992) or individual facts(Etzioni, Cafarella et al. 2005).

Semantically, named entities and concepts are related by means of taxonomic relationships. So, the way to go from the instance level to the conceptual level is by discovering taxonomic relationships. The most important precedent is (Hearst 1992), which proved their effectiveness to retrieve hyponym/hypernym relationships.

Pattern	Example
such NP as {NP,}* {and|or} NP	such countries as Poland
NP {,} such as {NP,}* {and|or} NP	cities such as Barcelona
NP {,} including {NP,}* {and|or} NP	capital cities including London
NP {,} specially {NP,}* {and|or} NP	science fiction films, specially Matrix
NP {,} (and|or) other NP	The Sagrada Familia and other churches

Table 4 Hearst patterns

However, the quality of pattern-based extractions can be compromised by the problems of de-contextualisations and ellipsis. For example, de-contextualisations can easily be found in sentences like “There are several newspapers sited in big cities such as *El Pais* and *El Mundo*”; without a more exhaustive linguistic analysis we might erroneously extract “El Pais” and “El Mundo” as instances of “city”. For the second case, due to language conventions, we can find a sentence like “teams such as Barcelona and Madrid”; in this case, the ellipsis of the words “Futbol Club” and “Club de Futbol Real” respectively could result in the incorrect conclusion that “Barcelona” and “Madrid” are subtypes of “teams” instead of “Futbol Club Barcelona” and “Club de Futbol Real Madrid”.

Another limitation of pattern-based approaches is the fact that they usually present a relatively high precision but typically suffer from low recall due to the fact that the patterns are rare in corpora(Cimiano, Handschuh et al. 2004). Fortunately, as it stated in section 3.1, this data sparseness problem will be tackled by exploiting the

Web (Buitelaar, Olejnik et al. 2004; Velardi, Navigli et al. 2005) as a corpus.

Finally, to sum up, our unsupervised pattern-based approach will combine Hearst Patterns (to construct Web search engine queries exploiting the Web as a corpus) and linguistic analysis (to detect the hyponym/hypernim relations in the retrieved documents). In this manner, the overall performance of the process will be improved.

3.2.3 Web-Scale statistics

In general, the use of statistical measures (e.g. co-occurrence measures) in knowledge related tasks for inferring the degree of relationship between concepts is a very common technique when processing unstructured text(Stephen Jose, Jack et al. 1993; Lin 1998). However, statistical techniques typically suffer from the sparse data problem (i.e. the fact that data available on words of interest may not be indicative of their meaning). So, they perform poorly when the words are relatively rare, due to the scarcity of data. This problem can be addressed by using lexical databases (Lee, Kim et al. 1993; Richardson, Smeaton et al. 1994) or with a combination of statistics and lexical information, in hybrid approaches (Jiang and Conrath 1997; Resnik 1999). In this sense, some authors (Brill 2003) have demonstrated the convenience of using a wide corpus in order to improve the quality of classical statistical methods. Concretely, in (Turney 2001; Keller, Lapata et al. 2002) methods to address the sparse data problem are proposed by using the hugest data source: the Web.

However, the analysis of such an enormous repository for extracting candidate concepts and/or statistics is, in most cases, impracticable. Here is where the use of lightweight techniques that can scale well with high amounts of information, in combination with the statistical information obtained directly from the Web, can represent a good deal. In fact, on the one hand, some authors (Pasca 2004) have enounced the need of using simple processing analysis when dealing with such a huge and noise repository like the Web; on the other hand, other authors (Cimiano, Handschuh et al. 2004; Etzioni, Cafarella et al. 2005; Cilibrasi and Vitányi 2006) have demonstrated the convenience of using Web search engines to obtain good quality and relevant statistics.

Relevant statistics can be achieved, for example, by using such measures as the Pointwise Mutual Information (PMI, Eq. (1)) (Church, Gale et al. 1991) or the Symmetric Conditional Probability (SCP) (Dias, Santos et al. 2006).

$$(1)\ PMI(a,b) = \log_2 \frac{\rho(ab)}{\rho(a)\rho(b)}$$

PMI statistically assesses the relation between two words (a, b) as the conditional probability of a and b co-occurring within the text. To exploit the characteristics of this measure in a Web environment the degree of relationship between a pair of

concepts can be measured through a combination of queries made to a Web search engine (involving those concepts and, optionally, their context). Queries are constructed using the logical query language (AND, OR, NOT…) provided by the search engine. Concretely, Eq. (2) computes the probability of the co-occurrence of two terms from the Web hit count provided by a search engine when querying each of the terms separately.

$$(2)\ PMI_{IR}(a,b) = \log_2 \frac{\frac{hits(a\ AND\ b)}{\#total\_webs}}{\frac{hits(a)}{\#total\_webs}\frac{hits(b)}{\#total\_webs}}$$

This score is derived from probability theory. Here, ρ (problem AND choice) is the probability that problem and choice co-occur. If problem and choice are statistically independent, then the probability that they co-occur is given by the product ρ(problem) ρ(choice). If they are not independent, and they have a tendency to co-occur, then ρ(problem AND choice) will be greater than ρ(problem) ρ(choice). Therefore the ratio between ρ(problem AND choice) and ρ(problem) ρ(choice) is a measure of the degree of statistical dependence between problem and choice. Since we are looking for the maximum score among a set of choices –or candidates-, we can drop ρ(problem) because it has the same value for all choices, for a given problem word, obtaining the final expression.

In this work, in order to provide a scalable solution, this measure will be used to score the relatedness among an Analysed Entity and its extracted Named Entities and to select which final annotation for a Named Entity is the best, taking into account all of its potential candidates (i.e., both the Named Entity relatedness with the Analysed Entity and the candidate's relevance for each NE will be evaluated against the whole Web). See sections 4.1.2 and 4.1.3.2.3.

3.3 Knowledge repositories

This section is divided in two parts. Section 3.3.1 explains what is an ontology and presents its main characteristics. Section 3.3.2 is about WordNet and it is exposed how to use this knowledge repository to obtain synonyms, hypernyms and hyponyms of a word.

3.3.1 Ontology basics

In (Studer, Benjamins et al. 1998), an ontology is defined as a formal, explicit specification of a shared conceptualization. Conceptualization refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon. Explicit means that the type of concepts used, and the constraints of their use, are explicitly defined. Formal refers to the fact that the

ontology should be machine-readable. Shared reflects the notion that an ontology captures consensual knowledge, that is, it is not private of some individual, but accepted by a group.

In (Neches, Fikes et al. 1991), a definition focused on the form of an ontology is given. 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. Other approaches have defined ontologies as explicit specifications of a conceptualization (Gruber 1995) or as shared understanding of some domain of interest (Uschold and Gruninger 1996).

From a formal point of view (Stumme, Ehrig et al. 2003; Cimiano 2006) an ontology has been defined as:

$$(3)\ O = (C, \leq_C, R, \sigma_R, \leq_R, A, \sigma_A, T) \text{ where,}$$

- C,R,A and T represent disjoint sets of concepts, relations, attributes and data types. Concepts are sets of real world entities with common features (such as different types of diseases, treatments, actors, etc.). Relations are binary associations between concepts. There exist inter-concept relations which are common to any domain (such as hyponymy, meronymy, etc.) and domain-dependent associations (e.g. an Actor performs an Action, a Disease is treated with a certain Treatment, etc.). Attributes represent quantitative and qualitative features of particular concepts (such as the medical code of a Disease, the degree of contagiousness, etc), which take values in a given scale defined by the data type (e.g. string, integer, etc.).
- ≤C represents a concept hierarchy or taxonomy for the set C. In this taxonomy, a concept c1 is a subclass, specialization or subsumed concept of another concept c2 if and only if every instance of c1 is also an instance of c2 (which represents its superclass, generalization or subsumer). Concepts are linked by means of transitive is-a relationships (e.g. if respiratory disease is-a disorder and bronchitis is-a respiratory disease, then it can be inferred that bronchitis is-a disorder). Multiple inheritance (i.e. the fact that a concept may have several hierarchical subsumers) is also supported (for example, Leukaemia may be both a subclass of Cancer and Blood disorder).
- ≤R represents a hierarchy of relations (e.g. has primary cause may be a specialization of the relation has cause, which indicates the origin of a Disorder).
- σR: R→C+ refers to the signatures of the relations, defining which concepts are involved in one specific relation of the set R. For example, the signature σ(is treated with): is treated with -> [Disease, Treatment] indicates that is_treated_with establishes a relation between the two concepts Disease and Treatment. It is worth to note that some of the

concepts in C+ correspond to the domain (the origin of the relation) and the rest to the range (the destination of the relation). In this example, Disease is the domain of the relation is_treated_with, and Treatment is the range. Those relationships may fulfil properties such as symmetry or transitivity.

- σA: A→CxT represents the signature describing an attribute of a certain concept C, which takes values of a certain data type T (e.g. the number of the leukocytes attribute of the concept Blood Analysis, which must be an integer value).

Different knowledge representation formalisms exist for the definition of ontologies. However, they share the following minimal set of components:

- Classes: represent concepts. Classes in the ontology are usually organised in taxonomies through which inheritance mechanisms can be applied.
- Relations: represent a type of association between concepts of the domain. Ontologies usually contain binary relations. The first argument is known as the domain of the relation, and the second argument is the range. Binary relations are sometimes used to express concept attributes. Attributes are usually distinguished from relations because their range is a data type, such as string, numeric, etc., while the range of a relation is a concept.
- Instances: are used to represent elements or individuals in an ontology.

Optionally, an ontology can be populated by instantiating concepts with real world entities (e.g. Saint John's is an instance of the concept Hospital). Those are called instances or individuals.

By default, concepts may represent overlapping sets of real entities (i.e. an individual may be an instance of several concepts, for example a concrete disease may be both a Disorder and a Cause of another pathology). If necessary, ontology languages permit to specify that two or more concepts are disjoint (i.e. individuals cannot be instances of more than one of those concepts).

Some standard languages have been designed to construct ontologies. They are usually declarative languages based on either first-order logic or on description logics. Some examples of such ontology languages are KIF, RDF, KL-ONE, DAML+OIL and OWL (Gómez-Pérez, Fernández-López et al. 2004). There are some differences between them according to their supported degree of expressiveness. In particular, OWL is the most complete one, allowing to define, in its more expressive forms (OWL-DL and OWL-Full) logical axioms representing restrictions at a class level. They are expressed with a logical language and contribute to define the meaning of the concepts, by means of specifying limitations regarding the concepts to which a given one can be related to. Several restriction types can be defined:

- Cardinality: defines that a concept's individual can be related (by means of a concrete relation type) to a minimum, maximum or exact number of other concept's instances. For example, certain types of Disease may have at

minimum one Symptom.

- Universality: indicates that a concept has a local range restriction associated with it (i.e. only a given set of concepts can be the range of the relation). For example, all the Symptoms of a certain Disease must be of the same type, the same concept category.
- Existence: indicates that at least one concept must be the range of a relation. For example a Disease always presents a certain kind of Symptoms, even though other ones may also appear.

All those restrictions can be defined as Necessary (i.e. an individual should fulfil the restriction in order to be an instance of a particular class) or Necessary and Sufficient (i.e. in addition to the previous statement, an individual fulfilling the restriction is, by definition, an instance of that class). This is very useful for implementing reasoning mechanisms when dealing with unknown individuals.

In addition, OWL also permits to represent more complex restrictions by combining several axioms using standard logical operators (AND, OR, NOT, etc.). In this manner, it is could be possible to define, for example, a set of Symptoms which co-occur for a particular Disease using the AND operator.

Considering the properties which ontologies have, they will be used in this work, on one hand, to drive the extraction process and to indicate what kind of features are relevant in a particular domain (i.e. only the important features for a particular domain will be annotated in the last step of the methodology avoiding an important computational cost annotating all the concepts which appear in the analysed text). On the other hand, ontology relations will be exploited in order to find taxonomically relations among classes, especially instance-concept relationships. These relationships will be useful to discover a set of potential concepts for a certain named entity.

3.3.2 WordNet, a generic knowledge repository

WordNet is a general purpose semantic electronic repository for the English language. In this section, an overview of its characteristics, structure and potential usefulness for our purposes is described.

WordNet2 is the most commonly used online lexical and semantic repository for the English language. Many authors have contributed to it (Daudé, Padró et al. 2003) or used it to perform many knowledge acquisition tasks. In more detail, it offers a lexicon, a thesaurus and semantic linkage between the major part of English terms. It seeks to classify words into many categories and to interrelate the meanings of those words. It is organised in synonym sets (synsets): a set of words that are interchangeable in some context, because they share a commonly-agreed upon

[2] http://wordnet.princeton.edu

meaning with little or no variation. Each word in English may have many different senses in which it may be interpreted: each of these distinct senses points to a different synset. Every word in WordNet has a pointer to at least one synset. Each synset, in turn, must point to at least one word. Thus, we have a many-to-many mapping between English words and synsets at the lowest level of WordNet. It is useful to think of synsets as nodes in a graph. At the next level we have lexical and semantic pointers. A semantic pointer is simply a directed edge in the graph whose nodes are synsets. The pointer has one end we call a source and the other end we call a destination.

Some interesting semantic pointers are:

- *hyponym*: X is a hyponym of Y if X is a (kind of) Y.
- *hypernym*: X is a hypernym of Y if Y is a (kind of) X.
- *part meronym*: X is a part meronym of Y if X is a part of Y.
- *member meronym*: X is a member meronym of Y if X is a member of Y.
- *attribute*: A noun synset for which adjectives express values. The noun weight is an attribute, for which the adjectives light and heavy express values.
- *similar to*: A synset is similar to another one if the two synsets have meanings that are substantially similar to each other.

Finally, each synset contains a description of its meaning, expressed in natural language as a gloss. Example sentences of typical usage of that synset are also given. All this information summarizes the meaning of a specific concept and models the knowledge available for a particular domain.

In this work, WordNet will be particularly useful to extract similar terms for a given term exploiting the hyponyms, hypernyms, and synsets. This will be beneficial in order to increase the set of candidates for a given Named Entity improving the matching process (see section 4.1.3.2.2). For example, Figure 5 shows the terms returned when querying the concept “church”. It shows the different meanings of church (polysemy) and using the aforementioned semantic pointers it can be determined that the term “church building” is a direct synonym of church, the terms abbey, basilica, cathedral, duomo and kirk are direct hyponyms, and the terms place of worship, house of prayer, house of God, house of worship are direct hypernyms.

Noun

S: (n) **church**, Christian church (one of the groups of Christians who have their own beliefs and forms of worship)

S: (n) **church**, church building (a place for public (especially Christian) worship) *"the church was empty"*

direct hyponym / *full hyponym*

S: (n) abbey (a church associated with a monastery or convent)

S: (n) basilica (an early Christian church designed like a Roman basilica; or a Roman Catholic church or cathedral accorded certain privileges) *"the church was raised to the rank of basilica"*

S: (n) cathedral (any large and important church)

S: (n) cathedral, duomo (the principal Christian church building of a bishop's diocese)

S: (n) kirk (a Scottish church)

part meronym

domain category

direct hypernym / *inherited hypernym* / *sister term*

S: (n) place of worship, house of prayer, house of God, house of worship (any building where congregations gather for prayer)

derivationally related form

S: (n) church service, **church** (a service conducted in a house of worship) *"don't be late for church"*

S: (n) **church** (the body of people who attend or belong to a particular local church) *"our church is hosting a picnic next week"*

Verb

S: (v) **church** (perform a special church rite or service for) *"church a woman after childbirth"*

Figure 5 Information extracted from WordNet when querying church

3.4 Conclusions

As seen in this chapter, the development of automatic and unsupervised solution needs an amount of techniques and technologies in order to obtain reliable results.

However, many classical knowledge acquisition techniques present performance limitations due to the typically reduced used corpus. Being unsupervised and domain-independent, it is needed a big corpus which represents the real distribution of information in the world in order to obtain reliable. Nevertheless, it does not exist such kind of repository, but as it has been stated in (Cilibrasi and Vitányi 2006), the amount and heterogeneity of information in the Web is so high that it can be assumed to approximate the real distribution of information in the world. For that reason, the Web has been proposed as a reliable work environment to minimize the problems of classical knowledge acquisition techniques.

Unfortunately, the Web is so huge that it is not possible to be analysed in a scalable way. For that, lightweight analyses, Web-based statistical measures and Web snippets have been introduced enabling the development of knowledge acquisition methodologies in a direct way.

In fact, as this work is focused on information extraction from any kind of Web

resource, including plain texts, it is also need a mechanism to interpret texts and, the concept of Natural Language Processing (NLP) has been introduced.

Moreover, as the extraction process is based on the detection of named entities (which represent real entities) and its annotation; it is needed a way to find the concepts which named entities represent and lexico-syntactic patterns, especially Hearst Patterns, have been proposed to carry out this task.

In this work, ontologies are used to drive the extraction process indicating the concepts that we want to extract from an analysed entity in a particular domain or area of study.

Finally, WordNet has been presented as a knowledge repository that can be used to extract synonyms, hypernyms and hyponyms of a word. This can be useful when the potential subsumer concepts of a named entity extracted by means of Hearst Patterns have not match with ontological classes and getting synonyms, hypernyms and hyponyms the probability of ontology matching increases.

4 Methodology

In this section the methodology implemented to achieve the goals of the work is presented. From a general point of view, the method consists of discovering relevant features about an analysed entity and matching these features with ontological concepts giving them semantic meaning. However, these must be applied in different kinds of Web resources (Structured and semi-structured) and must extract the relevant features in a domain independent way. This restriction will be achieved using domain ontologies to specify what kind of information is interesting for a particular area of study. For these reasons, a generic algorithm has been designed facilitating its application to different resources.

- In §4.1 the generic algorithm is described. It takes as input a Web-document to be analysed, a String that represents the analysed entity and a domain ontology which specifies the important concepts that should be extracted, and it returns, as a result, the relevant features (i.e. Named Entities) annotated semantically with concepts that appear in the input domain ontology.
- In §4.2 the applicability of the algorithm is studied in different kinds of resources. Specifically, plain text documents and Wikipedia articles have been taken into account.

4.1 Generic algorithm description

```
1  OntologyBasedExtraction(WebDocument wd, String AE,  DomainOntology do){
2      named_entity, sc, oc  is String
3      SC  is list of sc
4      soc  is record of {sc, oc}
5      SOC  is list of soc
6      ac  is soc
7      ne is record of {named_entity, SC, ac}
8      PNE is list of ne
9
```

```
10      /* Document Parsing */
11      pd ← parse_document(wd)
12
13      /* Extraction and selection of Named Entities from Document */
14      PNE ← extract_potential_NEs(pd)
15      ∀ pne_i ∈ PNE {
16        if NE_Score(pne_i, AE) > NE_THRESHOLD{
17          NE ← NE ∪ pne_i
18        }
19      }
20      /* Retrieving potential subsumer concepts for each NE*/
21      ∀ ne_i ∈ NE {
22        SC ← extract_subsumer_concepts(ne_i)
23        ne_i ← add_subsumer_concepts_list(SC)
24      }
25
26      /* Annotating NEs with ontological classes */
27      OC ← extract_ontological_classes(do)
28      ∀ ne_i ∈ NE {
29        /* Retrieving Subsumer Ontological Classes
30        (i.e. potential annotations) for each Subsumer Concept of each NE*/
31        SC ← get_subsumer_concepts_list(ne_i)
32        /* Applying direct matching */
33        SOC ← extract_direct_matching(OC, SC)
34        /* if no direct matching, Semantic matching is applied*/
35        if |SOC| == 0 {
36          SOC← extract_semantic_matching(OC, SC)
37        }
38        /* if a similar ontological class is found, the most proper
39        Annotation is chosen and the annotation is performed */
40        if |SOC| > 0 {
41          SOC ← SOC_Score(SOC, ne_i , AE)
42          ac ← select_SOC_wih_maxim_score(SOC, AC_TRESHOLD)
43          ne_i ← add_annotation(ac)
44        }
45      }
46      return NE
47    }
```

Algorithm 1 Generic algorithm for the implemented methodology

The previous algorithm (Algorithm 1) shows the main steps of our methodology. The key point of this algorithm is that it is generic, fact which implies that, overwriting some functions, it is possible analyse different types of documents (i.e. plain text documents, semi-structured documents or structured documents). Moreover, using different input ontologies the system is able to extract features of different domains, giving flexibility to the implemented method.

In order to discover the relevant features of an object, we focus on the extraction and selection of Named Entities (referred as NEs, see section 4.1.2) found in the text. It is assumed that NEs describe, in a way less ambiguous than general words, the relevant features of the analysed entity. A relevance-based analysis based on Web co-occurrence statistics is performed in order to select which of the NEs are the most related to (i.e., identify better) the analysed entity. Afterwards, the selected NEs are matched to the ontological concepts to which they could be considered as instances. In this manner the extracted features are presented in an annotated fashion, easing the posterior application of semantically-grounded data analyses.

The main steps will be explained in detail in the next subsections. In section 4.2 it is discussed how to take profit of two different types of resources (overwritting the aforementioned functions), namely plain text documents and semi-structured Wikipedia articles.

4.1.1 Document parsing

The first step is to parse a Web document (line 11) which is supposed to describe a particular real world entity, from now on Analysed Entity (AE). The Parse_document function depends on the kind of document that is being analysed. If this is an HTML document, then it is necessary to extract raw text from it by means of HTML parsers which are able to drop headers, templates, HTML tags, etc. Otherwise, if the document is a semi-structured source such as Wikipedia article, then other tools are used in order to filter and select the main text.

4.1.2 Named Entities

This step consists in extracting relevant named entities from the analysed document. NEs represent real world entities. In other words, named entities can be considered as instances of ontological concepts (Berners-Lee, Hendler et al. 2001) (e.g. Tarragona is an instance of a city).

The function extract_potential_NEs (line 14) returns a set of Named Entities (PNE) but only a subset of the elements of PNE describes the main features of AE; the rest of the elements of PNE introduce noise because they are not directly related to the analysed entity (they just happen to appear in the Web page describing the entity but are not part of its basic distinguishing characteristics). Thus, it is necessary to have a way of separating the relevant NEs from the irrelevant ones (NE filtering,

line 16). To do that, we use a Web-based co-occurrence measure that tries to assess the degree of relationship between AE and each NE. In fact, it has been stated that the amount and heterogeneity of information in the Web is so high that it can be assumed to approximate the real distribution of information in the world (Cilibrasi and Vitányi 2006). Concretely, a version of the Pointwise Mutual Information (PMI, stated in 3.2.3) relatedness measure adapted to the Web is computed (Church, Gale et al. 1991).

$$(4)\ NE_{SCORE}(PNE_i, AE) = \frac{hits(PNE_i\ AND\ AE)}{hits(PNE_i)}$$

In the NEscore (Equation (4)), concept probabilities are approximated by Web hit counts provided by a Web search engine. Finally, the NEs that have a score exceeding an empirically determined threshold (NE_THRESHOLD, line 16) are considered as relevant, whereas the rest are removed. The value of the threshold will determine a compromise between the precision and the recall of the system.

4.1.3 Semantic Annotation

The aim of semantic annotation, in this work, is to match features with the appropriate ontology classes.

In this area, some approaches have been proposed. One way to assess the relationship between two terms (which, in our case, would be a NE and an ontology class) is to use a general thesaurus like WordNet to compute a similarity measure based on the number of semantic links among the queried terms (Wu and Palmer 1994; Leacock and Chodorow 1998). However, those measures are hampered by WordNet's limited coverage of NEs and, in consequence, it is usually not possible to compute the similarity between a NE and an ontological class in this way.

There are approaches which try to discover automatically taxonomic relationships (Sanderson and Croft 1999; Bisson, N\'dellec et al. 2000), but they require a considerable amount of background documents and linguistic parsing.

Finally, another possibility is to compute the co-occurrence between each NE and each ontological class using Web-scale statistics as the relatedness measure (Turney 2001), but this solution is not scalable because of the huge amount of required queries (Cimiano, Ladwig et al. 2005).

We will use the last technique, but introducing a previous step that reduces the number of queries to be performed.

So, in our approach the semantic matching is divided in two parts: the discovery of potential subsumer concepts (line 22) and their matching with the ontology classes (lines 27-46).

The first part is proposed in order to minimize the number of queries (NE, ontology class) to be performed in which ontology classes that are potentially good

candidates for the matching are discovered. If the number of candidates is small, it will feasible to use Web-scale statistics to compute the relatedness between them and the NE. It may be noticed that the problem is finding a bridge between the instance level (i.e., a NE) and the conceptual level (i.e. an ontology concept for which the NE is an instance). Semantically, NEs and concepts are related by means of taxonomic relationships. So, the way to go from the instance level to the conceptual level is by discovering taxonomic relationships. Taxonomic relations are considered as subsumer concepts. Notice that those concepts are abstractions of the NE and they not depend on any ontology. This means that subsumer concepts needn't match with ontological classes but they can.

The second part tries to match the found subsumer concepts with ontological classes, if it is possible.

4.1.3.1 Discovering potential subsumer concepts

The first task of semantic annotation consists in discovering potential subsumer concepts for each relevant named entity (line 22).

Subsumer concepts are abstractions of collections of real entities which share common characteristics among them. For example, the subsumer concept of the real entity *The Sagrada Familia* or *St. Peter's Basilica* is *basilica*. Notice that real entities may belong to different concepts such as basilica and monument. Other important characteristic of subsumer concepts is that they can be represented by different terms which are equivalent. Consider, for instance, *Porsche* such a real entity, where its subsumer concept could be *car*, *automobile*, *auto*, *motorcar* and *machine*. Finally, the abstraction can be performed in different levels. In the case of *the Sagrada Familia* its direct subsumer is *basilica* but higher subsumer concepts such *roman building* and *religious building* can be considered.

By means of the function extract_subsumer_concepts a set of potential subsumer concepts for each NE is extracted. Then, the last step of the methodology aims to find a correspondence between the potential subsumers of each NE and the classes of an ontology. We use an input ontology in order to drive the extraction process and to indicate what kinds of features are relevant in a particular domain.

4.1.3.2 Ontology matching

We distinguish between two types of matching: Direct Matching and Semantic Matching. Moreover, there are situations in which there is evidence that a certain NE is related to several ontological classes. In this case, Web-based statistical measures are applied again in order to choose the most representative one (Class Selection, section 4.1.3.2.3). These three steps are explained in the following subsections.

4.1.3.2.1 Direct Matching

In this initial step, the system tries to find a direct match between the potential

subsumers of a NE and the ontology classes. This phase begins with the extraction of all the classes contained in the domain ontology (line 27). Then, for each Named Entity NE_i, all its potential subsumer concepts (SC_i) are compared against each ontology class in order to discover the most similar ontological classes (SOC_i, line 31-33), i.e., classes whose name matches the subsumer itself or a subset of it (e.g., if one of the potential subsumers is "Gothic cathedral"', it would match an ontology class called "Cathedral"). A stemming algorithm is applied to both SC_i and ontology classes in order to discover terms that have the same root (e.g., "city" and "cities"). If one (or several) ontology classes match with the potential subsumers, they are included in SOC_i as candidates for the final annotation of NE_i. This direct matching step is quite easy and computationally efficient; however, its main problem is that, in many cases, the subsumers do not appear as ontology classes with exactly the same name, and potentially good candidates for annotation are not discovered.

4.1.3.2.2 Semantic Matching

The semantic matching (line 36) step is performed when the direct matching has not produced any result (line 35).

Its main goal is to increase the number of elements in SC_i, so that the direct matching can be tried again with a wider set of terms. The new potential subsumers are concepts semantically related to any of the initial subsumers (synonyms, hypernyms and hyponyms). As we are working at a conceptual level, WordNet has been used to obtain these related terms and to increase the set SC_i. The main problem of semantic matching is that many words are polysemous and, before extracting the related concepts from WordNet, we have to discover which is the synset that corresponds with the intended sense of the word in the domain (i.e., a semantic disambiguation step must be performed).

One of the main problems when analysing natural language resources is semantic polysemy. For example, if the primary keyword has more than one sense (e.g. virus can be applied over “malicious computer programs” or “infectious biological agents”), the resulting ontology may contain concepts from different domains (e.g. “iloveyou virus”, “immunodeficiency virus”). This problem is generally known as word sense disambiguation and has proved to be more difficult than syntactic disambiguation.

The meaning of a word in a particular usage can only be determined by examining its context. This is, in general, a trivial task for the humans, but the task has proved to be difficult for computer.

In order to deal with sense disambiguation, it is proposed a Web-based approach combining the context from a Named Entity has been extracted, WordNet definitions and cosine distance.

Thus, the first step is, for each element of SC_i of each NE_i, look it up in WordNet. If it only has one definition (synset), the new subsumer candidates (synonyms, hypernyms and hyponyms) are retrieved. Otherwise, if the element of

SC_i has more than one synset, it is necessary to choose the most suitable one (word sense disambiguation).

One possible solution is to use the context (i.e., the sentence from which NE_i was extracted) but, usually, this context is not enough to disambiguate the meaning.To minimize this problem, the Web is used again in order to extract new evidences of the relationship between NE_i and AE. A Web query containing AE and NE_i is performed, and a certain number of snippets are retrieved. Then, the system calculates the cosine distance between each snippet and all the synsets of the element of SC_i. The synset with a higher average value is finally selected.

Next, an example of the method is explained. Table 5 depicts the input data of the problem. First three rows are the analysed entity, the named entity and its subsumer concept. The rest of the data indicates the performed query in order to retrieve web snippets and all the WordNet synsets for the subsumer concept.

Data	value
AE:	Barcelona
NE_i:	Sagrada Familia
SC_i:	Cathedral
Query:	"Barcelona" + "Sagrada Familia"
Synset 1:	[cathedral] any large and important church
Synset 2:	[cathedral, duomo] the principal Christian church building of a bishop's diocese

Table 5 Semantic disambiguation example (part 1)

Table 6 shows a subset of the retrieved snippets, which represent the new context, and the final score when applying cosine distance between the context and the synset. As a final result, synset 1 obtains the highest score and synonyms, hyponyms and hypernyms are extracted from it. In this example, the related terms for the subsumer concepts retrieved from WordNet are: minster, church and church building.

Snippet/context	Synset 1	Synset 2
- His best known work is the immense but still unfinished church of the Sagrada Família, which has been under construction since 1882, and is still financed by private donations.	0.16	0.11
- Review of Barcelona's greatest building the Sagrada Familia by Antonio Gaudi, Photos, and links	0.0	0.12
- The Sagrada Familia is the most famous church in Barcelona ... As a church, the Sagrada Familia should not only be seen in the artistic point of view	0.26	0.18
- The Sagrada Familia (Holy Family) is a church in Barcelona, Spain. ... The architect who designed the Sagrada Familia is Antoni Gaudí, the designer of more other ...	0.12	0.08
- Virtual Tour of Barcelonas's sightseeings. ... commonly known as the Sagrada Família, is a large Roman Catholic church in Barcelona, Catalonia, Spain, ...	0.28	0.10
[...]	[...]	[...]

Table 6 Semantic disambiguation example (part 2)

4.1.3.2.3 Class Selection

When more than one ontology class has been proposed (line 40) as annotation for a certain NE_i, the final step is to choose the most appropriate one. The selection is based on the relatedness between the Named Entity and each element of SOC_i, assessed again with the Web-based version of PMI introduced in section 3.2.3. However, it must be noted that the elements of SOC_i can also be polysemous, and can be referring to different concepts depending on the context (line 41). So, in Eq. (5), the analysed entity AE has been introduced to contextualize the relationship of each element of SOC_i with NE_i.

$$(5)\ SOC_{SCORE}(SOC_{ij}, NE_i, AE) = \frac{hits(AE\ \&\ NE_i\ \&\ SOC_{ij})}{hits(AE\ \&\ SOC_{ij})}$$

The score (E.q. (5)) computes the probability of the co-occurrence of the named entity NE_i and each ontology class proposed for annotation SOC_{ij} from the Web hit count provided by a search engine when querying these two terms (contextualized with *AE*). Finally, only the annotation with the highest score which reaches the AC_Treshold (line 42) is annotated

4.2 Applying the algorithm to different types of Web resources

So far, the generic feature extraction algorithm has been presented. This section discusses which functions should be overwritten in order to apply it to different types of resources. In order to demonstrate its applicability, this work is focused in two types of resources: plain texts (unstructured resources) and Wikipedia articles (semi-structured resources). Particularly, the functions that have to be overwritten are extract_potential_NEs (line 14) and extract_subsumer_concepts (line 22). The rest of the steps do not depend on the kind of resource and the generic algorithm is applied. In following sections both cases are presented.

4.2.1 Extraction from raw text

The extraction process from raw text (i.e. plain text) is the most difficult task. For that reason, in this section, it is presented how to deal with the main problems that arise from it. Notice that this kind of repositories are the most extended around the Web and for that reason it is very important to have a mechanism to exploit all the available information.

4.2.1.1 Named Entities detection

The main problem related with NE detection is the fact that they are unstructured and unlimited by nature as is stated in (Sánchez, Isern et al. 2010). This implies that, in most cases, these NEs are not contained in classical repositories as WordNet due to its potential size and its evolvability.

Different approaches in the field of NE detection have been proposed. Roughly, they can be divided into supervised and unsupervised methods.

Supervised approaches try to detect NEs relying on a specific set of extraction rules learned from pre-tagged examples (Stevenson and Gaizauskas 2000; Fleischman and Hovy 2002), or predefined knowledge bases such as lexicons and gazetteers (Mikheev and Finch 1997). However, the amount of effort required to assemble large tagged sets or lexicons binds the NE recognition to either a limited domain (e.g., medical imaging), or a small set of predefined, broad categories of interest (e.g., persons, countries, organizations, products). This introduces compromises in the recall (Pasca 2004).

In unsupervised approaches like (Lamparter, Ehrig et al. 2004), it has been proposed to use a thesaurus as background knowledge (i.e., if a word does not appear in a dictionary, it is considered as a NE). Despite the fact that this approach is not limited by the size of the thesaurus, misspelled words are wrongly considered as NEs whereas correct NEs composed by a set of common words are rejected, providing inaccurate results.

Other approaches take into consideration the way in which NEs are presented in the specific language. Concretely, languages such as English distinguish proper names from other nouns through capitalization. The main problem is that basing the detection of NEs on individual observations may produce inaccurate results if no additional analyses are applied. For example, a noun phrase may be arbitrary capitalised to stress its importance or due to its placement within the text. However, this simple idea, combined with linguistic pattern analysis, as it has been applied by several authors (Hahn and Schnattinger 1998; Cimiano, Handschuh et al. 2004; Pasca 2004; Downey, Broadhead et al. 2007), provides good results without depending on manually annotated examples or specific categories.

Being unsupervised, domain-independent and lightweight, in this work, the last approach has been implemented, as follows, in order to detect NEs.

First, the four modules of the OPENNLP parser (Sentence Detector, Tokenizer, Tagging and Chunking) are applied in order to analyse syntactically the input text of the Web document. The last module is able to tag Proper Nouns, which represent NEs, using an internal database, but this approach produces a low recall because of the reasons stated in section 4.1.2. For example, in *[NP The/VB gothic/JJ cathedral/NN][VP of/VB][NP Barcelona/NNP]*, the noun phrase (NP) *Barcelona* is tagged as proper noun (/NNP) but, in *[NP Tarragona/EX][VP is/NNS][NP a/JJS city/NN]*, NP *Tarragona* is erroneously not considered as proper noun. To avoid a

supervised methodology based on a database, the output of OPENNLP has been complemented by capitalization heuristics where all Noun Phrases which contain one, or more than one, word begins with a capital letter has been considered as a NE and consequently a set of potential Named Entities (PNE) is detected. Thus, that all Noun Phrases which contain at least one word that begins with a capital letter are considered as a potential NE.

Table 7 shows an example of the extracted NE from the first fragment of text of Wikipedia article about Tarragona. Notice that only Spain is detected as a proper noun using only the natural language parser but applying capitalization heuristics the rest of Named Entities has been extracted.

Detected sentences	Extracted NE	Correct?
[NP Tarragona/EX]	Tarragona	ok
[NP Catalonia/NN]	Catalonia	ok
[NP Spain/NNP]	Spain	ok
[NP Sea/NNP]	Sea	ko
[NP Tarragonès/VBZ]	Tarragonès	ok
[NP the/VBZ Vegueria/NNPS]	the Vegueria	ko

Table 7 Set of extracted NE from Tarragona Wikipedia introduction

4.2.1.2 Discovering potential subsumer concepts

We use the standard Hearst's taxonomic linguistic patterns, which have proved their effectiveness to retrieve hyponym/hypernim relationships (Hearst 1992). We exploit the Web as the corpus from which to extract the semantic evidences of the appearances of the patterns (Rozenfeld and Feldman 2008). The main reason of using the Web as the corpus is because of the fact that explicit linguistic patterns are difficult to find in reduced corpora, that normally offer a relatively high precision but suffer from low recall.

The system constructs a Web query for each NE and for each pattern. Each query is sent to a Web search engine, which returns as a result a set of Web snippets. Finally, all these snippets are analysed in order to extract a list of potential subsumer concepts (i.e., expressions that denote concepts of which the NE may be considered an instance).

Pattern structure	Query	Example
CONCEPT such as NE	"such as Barcelona"	*cities* such as Barcelona
such CONCEPT as NE	"such * as Spain"	Such *countries* as Spain
NE and other CONCEPT	"Ebre and other"	Ebre and other *rivers*
NE or other CONCEPT	"The Sagrada Familia or other"	The Sagrada Familia or other *monuments*
CONCEPT especially NE	"especially Tarragona"	*World Heritage Sites* especially Tarragona
CONCEPT including NE	"including London"	*capital cities* including London

Table 8 Patterns used to retrieve potential subsumer concepts

Table 8 summarizes the linguistic patterns that have been used (CONCEPT represents the retrieved potential subsumer concept and NE the Named Entity that is being studied).

4.2.2 Extraction from semi-structured Wikipedia documents

As stated in section 3.1.3, Wikipedia provides some particularities, which can be useful when extracting information. Specially, this work is focused on *internal links* and *category links*. The first ones represent connections among terms that appear in a Wikipedia article with other articles, which are talking about the aforementioned terms. *Category links* group different articles in areas that are related in some way and give articles a kind of categorization.

4.2.2.1 Named Entities detection

In order to take profit of links structure, internal links have been considered as potential named entities (PNE). The hypothesis is that internal links have been created by a big community of users and it can be assumed that the information which they represent has been revised for enough readers (of which some of them may be experts of the topic that the article is about) to assume that it is correct.

The problem of PNE extracted from internal links are that, on one hand, not all of them are directly related with the analysed entity (AE) and, on the other hand, only a subset of PNE are real NE.

In order to illustrate these problems, the following fragment of text extracted from Wikipedia will be examined. **"**Barcelona is the capital and the most populous city of Catalonia and the second largest city in Spain, after Madrid, with a population of 1,621,537 within its administrative limits on a land area of 101.4 $km^{2"}$. In this text, there are four terms linked with other Wikipedia articles by means of internal links. Three of them are NE (Catalonia, Spain and Madrid) and they represent instances of things, the other one is a common noun which represents a concept (capital). The first wikilink is not a NE because the first part of the sentence is defining what the NE Barcelona is, and the person who edited the article considered that the term "capital" (which represents a concept) was important for a correct understanding of the text. Finally, the NE Madrid is bringing information of general purpose that is not directly related with Barcelona and, in consequence, it is not a relevant feature for describing the entity Barcelona.

Due to these problems, the set of extracted PNE has to be filtered by means of the NE score presented in the generic algorithm. But, in this manner, the semi-structure of Wikipedia links provides a degree of reliability and it helps to avoid the problem of analysing plain text using NLP.

Table 9 shows the first NE detected when using wikilinks. It is important to observe that this step is only concerning with detection and this NE will be filtered

in next algorithm step. Notice that in "Barcelona Cathedral" and "Barcelona Pavilion" both cathedral and pavilion are common nouns but they are preceded by Barcelona specifying that it is referring a concrete real entity (i.e., a named entity).

Wikilinks	Correct?
Acre	ko
Antoni Gaudí	ok
Arc de Trionf	ok
Archeology Museum of Catalonia	ok
Barcelona Cathedral	ok(*)
Barcelona Museum of Contemporary art	ok
Barcelona Pavilion	ok(*)
Casa Batlló	ok

Table 9 Subset of extracted NE from Barcelona Wikipedia article

4.2.2.2 Discovering potential subsumer concepts

In order to extract potential subsumer concepts for each named entity Wikipedia category links have been used. As stated in section 3.1.3, category links have some attractive characteristics but present some limitations. They are useful because they classify in a kind of hierarchy all the articles which Wikipedia contains. This classification categorizes all the concepts and named entities in Wikipedia. This means that a wiki which is referring to a real entity belongs to one or more Wikipedia categories which in turn are included in higher categories.

Remember the example where "The Sagrada Familia" article was categorized as Antoni Gaudí buildings, Buildings and structures under construction, Churches in Barcelona, Visitor attractions in Barcelona, World Heritage Sites in Spain, Basilica churches in Spain, etc. Apparently, these categories are too complex to be used as subsumer concepts (i.e. it is not probable that a category matches directly with ontological classes) and some previous analysis is needed. So, the key concepts of each category have to be detected. For example in "Churches in Barcelona" the key concept is "Churches" and in "Buildings and structures under construction" there are two important concepts: "Buildings" and "Structures". To extract the main concepts of each sentence a natural language parser has been used, and all the Noun Phrases have been extracted.

Another limitation of Wikipedia categories is the fact that they do not always contain enough concepts to perform the matching among them and ontological classes. For instance, the NE *Plaça de Catalunya* is a square situated in the city centre of Barcelona. Its categories are *Plazas in Barcelona*, *the Eixample* and *Central business districts* but our ontology is focused on tourism domain and none of these concepts appears in it. By contrast, *Plaça de Catalunya* is considered as a tourist attraction in Barcelona and the concept *visitor attraction* is represented by the ontology. Fortunately, as mentioned before, Wikipedia categories are included in higher categories. Following the same example, *Plazas in Barcelona* belongs to the higher categories *Squares and plazas by city*, *Geography of Barcelona* and *Visitor*

attractions in Barcelona. Notice that the last one represents the same concept that we want to find and in consequence a new potential subsumer concept has been found. Observe that higher levels of categories represent higher concepts and going up through categories the meaning of the NE which they represent could be lost. Moreover, the Wikipedia categorisation has been performed by hand and its structure is approximated by a directed acyclic graph fact which implies that it is not always possible to navigate through categories in a taxonomical way. For that reason, only two levels of categories have been used in our approach.

Finally, the last limitation of Wikipedia categories is that sometimes they are composed by named entities and as we are looking for concepts they are not useful to extract potential subsumer concepts. For example, one of the categories of *Plaça de Catalunya* was *Eixample*, the name of a district of Barcelona.

Table 10 exemplifies the subsumer concepts extractions from Wikipedia categories. This is only a temporary list of potential subsumer concepts but they will be selected as potential subsumer concepts when applying ontology matching.

Wikilinks	**Potential subsumer concepts**
Antoni Gaudí	1852 births, 1926 deaths, architects, roman catholic churches, art nouveau architects, expiatori de la sagrada família, catalan architects, spanish ecclesiastical architects, modernisme architects, 19th century architects, 20th century architects, organic architecture, people, reus...
Arc de Trionf	triumphal arches, gates, moorish revival architecture, 1888 architecture, public art stubs, 1888 works, architecture, architecture, public art, public art, art stubs…
Archeology Museum of Catalonia	museums, archaeology museums, Sants-Montjuïc
Barcelona Cathedral	cathedrals, churches, visitor attractions, basilica churches…
Barcelona Museum of Contemporary art	museums, art museums, galleries, modern art museums, modernist architecture, spots, richard meier buildings, el raval, modern art…
Barcelona Pavilion	...
Casa Batlló	visionary environments, modernisme, antoni gaudí buildings, 1907 architecture, world heritage sites, spain, visitor attractions, eixample, passeig, gràcia, outsider art, 1907 works, 1900s architecture, edwardian architecture...

Table 10 Subset of extracted potential subsumer for Barcelona NEs

To summarize, Wikipedia categories give information that is usually composed by concepts and the relations represented by means of category links can be taxonomical, lexico-syntactic, semantics, synonyms, etc. So, categories can be used to extract subsumer concepts but applying some techniques to extract the key concepts of each category and following some restrictions.

4.2.3 Computational cost

The computational cost of the proposed method depends on the number of queries performed because they are the most expensive task (Cimiano, Ladwig et al. 2005). We can distinguish five different tasks in which queries are performed: NE detection, NE filtering, subsumer concepts extraction, semantic disambiguation and class selection.

Both plain text and semi-structured text analyses have the same cost for NE filtering, semantic disambiguation and class selection. On one hand, to rank NEs for the relevance filtering step, two queries are needed for each NE (i.e., 2n, where n represents the number of NEs). On the other hand, class selection requires as many queries as candidates a NE has (i.e. n(c/n)2, where c is the total number of candidates). For semantic disambiguation only one query is needed for each candidate (i.e. c).

So, the difference in computational cost between plain text analyses and semi-structured ones is in NE detection and subsumer concepts extraction. In the first approach six queries are performed to discover subsumer concepts by means of Hearst Patterns (6n). In the second approach, no queries are needed because NEs are directly extracted from the tagged text.

Thereby, the number of queries needed to analyse plain text is 8n+3c, whereas only 2n+3c are needed when dealing with Wikipedia articles. This shows how the exploitation of Wikipedia's structure aids to improve the performance of the method.

4.3 Conclusions

In this section, the main steps of our approach have been presented. First, a generic algorithm has been proposed. Being the algorithm generic, different kinds of resources can be analysed in order to extract relevant features of a studied real entity. The approach is focused on detecting named entities and annotating them, if possible, with concepts defined in domain ontologies. Only named entities are taken into account because they describe, in a way less ambiguous than general words, the most relevant features of the analysed entity.

To demonstrate the applicability of this generic algorithm, two types of resources have been studied. On one hand, it has been explained how to use the algorithm to extract information from plain texts which are unstructured and the most common resources in the World Wide Web. On the other hand, Wikipedia has been used as an example of semi-structured resource and it has been stated one approach to take profit of its links structure and categorization.

5 Evaluation

In this chapter some evaluation results are presented. The evaluation consists in three different parts that, study the influence of thresholds, considered Web resources and, the input domain ontology. The reason of using them as a subject of study is because the fact that they are the input parameters which can be set to adjust the algorithm behaviour, getting different levels of precision and recall.

The precision and recall have been computed in all tests. In order to calculate them, a domain expert has manually selected which of the features included in the articles are relevant for the subject of study (i.e. the analysed entity, AE) and which concepts in the ontology are the more adequate to annotate them if it is possible.

The recall is calculated by dividing the number of correct annotations performed by the system by the total of annotations the system should have annotated according to the expert's opinion (Equation (6)).

$$(6)\ Recall = \frac{\#\text{Good}_{\text{annotations}}}{\#\text{Good}_{\text{annotations}} + \#\text{Unretrieved\_Good}_{\text{annotations}}}$$

The Precision is the number of correct annotations according to the expert's opinion divided by the total number of annotations (Equation (7)).

$$(7)\ Precision = \frac{\#\text{Good}_{\text{annotations}}}{\#\text{Good}_{\text{annotations}} + \#\text{Bad}_{\text{annotations}}}$$

All the evaluations have been presented in the domain of tourism taking into account the specifications of DAMASK project. The rest of the section is structured as follows:

- In §5.1, two ontologies used to test the system are presented.
- The influence of the thresholds used in the algorithm is studied for the city of Barcelona using one ontology, in §5.2.
- §5.3 evaluates the behaviour of our methodology using different domain ontologies.
- §5.4 shows a comparison between the analysis of a plain text versus that of a semi-structured Wikipedia document.

- Finally, in §5.5, some conclusions are extracted and the main advantages and drawbacks of each method are discussed.

5.1 Used ontologies

The evaluation process has been performed using different ontologies to prove the applicability of the algorithm for different information to be extracted. In the following paragraphs, the ontologies used to carry out the tests are briefly described.

TourismOWL.owl ontology

This ontology models touristic points of interest for different kinds of tourist profiles. It was designed in a final year project (Vicient 2009) based on information extraction through Wikipedia articles. It consists of 315 classes and a depth of 5 hierarchical levels. Its main classes map concepts related with administrative divisions (borough, city, country, village, etc.), buildings (commercial buildings, cultural buildings, religious buildings, sport buildings, etc.), festivals (art festivals, music festivals, carnival, etc.), landmarks (commemorate landmarks, geographical landmarks, memorial landmarks, etc.), museums (archaeology museum, history museum, science museum, etc.) and sports (football, basketball, hockey, formula one, etc.). See Annex I – TourismOWL.

Space.owl ontology

This ontology was found looking up ontologies using the Web search engine SWOOGLE that is specialized in ontologies. The *Space.owl* ontology consists of 188 classes and a depth of 6 hierarchical levels. It maps concepts related with three main topics: geographical features (i.e., archipelago, beach, river, forest, etc.), geopolitical entities (i.e., country, capital, city, district, street, etc.) and places which includes business places (factory, convention centre, etc.), private places (residential structure, home, etc.) and, public places (educational and medical structures, entertainment places, shopping facilities, transportation connections, etc.). See Annex II – Space.owl.

5.2 Influence of thresholds

In this section the influence of the threshold for filtering the named entities (T1) and the threshold for selection annotation (T2) have been studied. The analysed entity for this test has been the Wikipedia article about Barcelona, using the *space.owl* ontology as input. The comparison between T1 and T2 has been performed taking the Wikipedia article as plain text and also as a semi-structured Web resource.

Figure 6 shows a comparison between both thresholds. In the left column, the comparison is applied for Barcelona taking as input the plain text of the article, while the right column presents the results when taking profit of Wikipedia semi-structure.

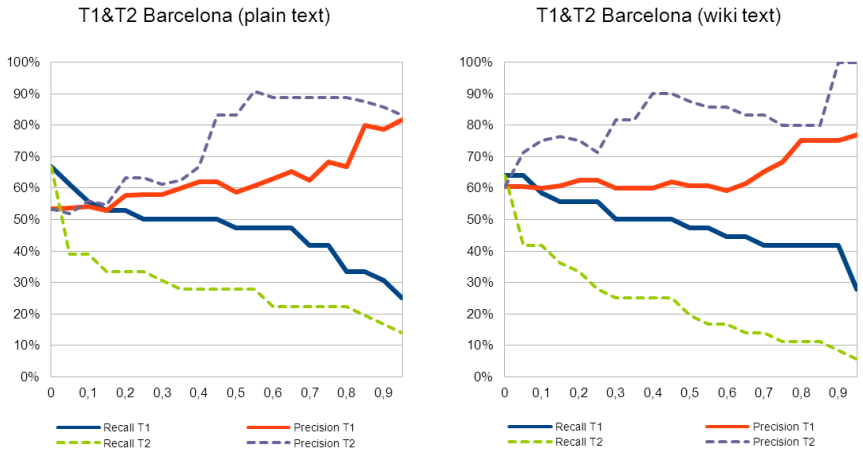


Figure 6 Influence of T1 and T2

The results show that the method is able to reach higher precisions with T2 but punishing the recall even more than T1. Notice that T1 is calculated taking as parameters the potential named entity and the analysed entity measuring the level of relatedness between both, but T2 goes further measuring the relatedness between the analysed entity, the potential named entity and the subsumer candidate to be annotated. This fact implies that the second threshold is more restrictive because the relatedness involves three parameters instead of two. Moreover, it is important to stress the fact that T2 has a double function: 1) it measures the relatedness degree between the named entity and its subsumer candidate facilitating the final annotation at the moment of chosing the best of the subsumer concepts for each named entity and 2) it contextualizes the ontology annotation in the domain of the analysed entity which implies that is performing a kind of named entity filtering like T1. However, T1 is necessary to decrease the number of Web queries because using only T2 the amount of those will be higher because each named entity usually has a high number of subsumer concepts, especially when analysing plain text resources and extracting the subsumer concepts by means of *Hearst Patterns*.

5.3 Plain text vs. Wikipedia document

In this second test, we picked up as case studies the Barcelona and Canterbury Wikipedia articles, which describe these cities. Final feature annotations were performed taking into account the *space.owl* ontology. The evaluation was performed by analysing the articles both as plain text and also taking profit of Wikipedia semi-structure. So, in both cases the analysed content was the same.

Figure 7 shows a comparison between the two methods (plain text and semi-structured) when applied to the cities of Barcelona and Canterbury. In the left column, the influence of the threshold for filtering NE (T1) and the threshold for selection annotation (T2) are studied for Barcelona, while the right column depicts the analysis for Canterbury.

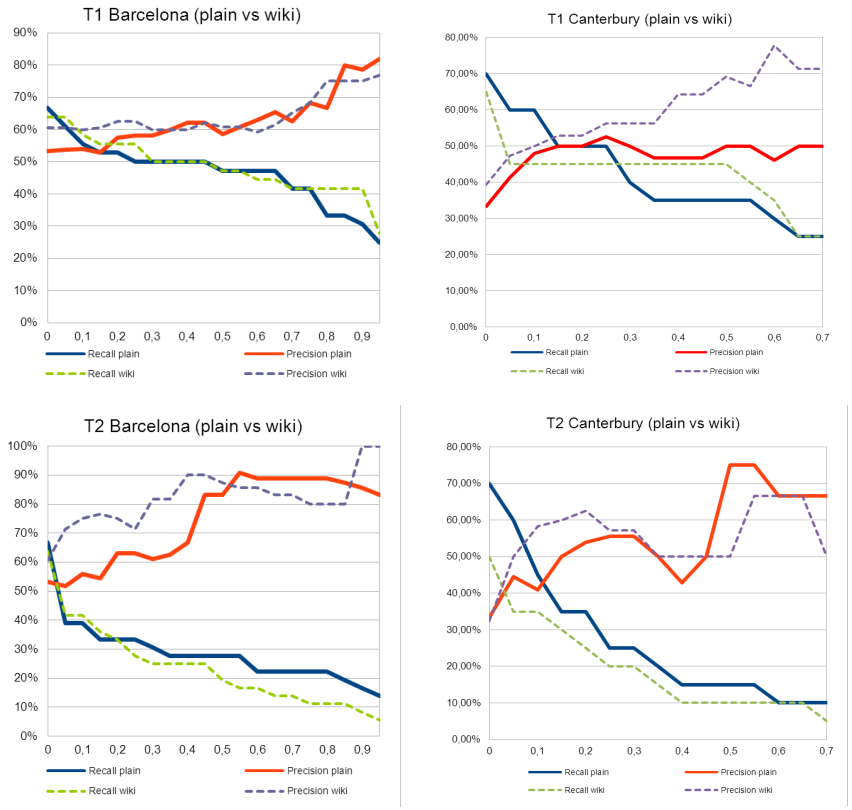


Figure 7 Plain text vs Wikipedia documents

The results show that the method is able to extract, in both cases, more than a 50% of the features marked for the domain expert. We also observe that the precision tend to keep or to improve when taking into account the semi-structure of the Wikipedia articles while the recall decreases. This is because in the first approach the whole textual content is analysed. This implies that there are more possibilities to detect representative features whereas the precision may be lower because there is a higher amount of unrepresentative features which add noise to the final results. On the opposite, using the second approach the set of analysed entities is limited to those manually annotated but, in contrast, the precision is higher because the potential candidates for each feature are extracted from Wikipedia categories (tagged and selected manually by a big community of users). It is important to note that, in any case, the analysis of Wikipedia articles is, as discussed in section 4.2.3, considerably faster than text, as the degree of analysis required to extract and annotate entities is reduced.

Considering that the final goal of the method is to enable the application of data analysis methods (such as clustering) a high precision would be desirable, even at the cost of a reduced recall. In these cases, selection thresholds can be tuned for a high precision establishing a more restrictive value.

5.4 Influence of domain ontologies

This section compares the performance of our method using different domain ontologies. In both cases, the same Wikipedia article for the city of Barcelona has been tested using the ontologies stated above in section 5.1.

Figure 8 shows the results of the evaluation. All the graphs compare the recall and precision reached for the algorithm when applying it with different input ontologies (*space.owl* and *tourismOWL.owl*). The left column depicts the results when analysing the Wikipedia article as plain text while the right column represents the semi-structured approach. Although the objective of this evaluation is not to study the influence of thresholds, both analyses are represented based on the values of T1 and T2. This is because, in this way, it is possible to observe the global trend of analyses instead of fixing two values for both thresholds.

If we observe the left column, we can see that for both thresholds the precision obtained with the tourism ontology is higher. However, this fact does not happen, in the right column where the precision and recall for both ontologies is similar. This is because the tourism ontology was created based on the text of different Wikipedia articles about cities focused on touristic activities. This means that, for the first column, there are more direct matches than for the second one. Even though the difference between precision and recall thresholds is not high when using both ontologies, we can see that the results are a bit better

for tourismOWL.owl ontology. These results show the importance of using a domain ontology proper to model the key concepts about the domain which is being studied in order to maximize the quality of the extracted features.

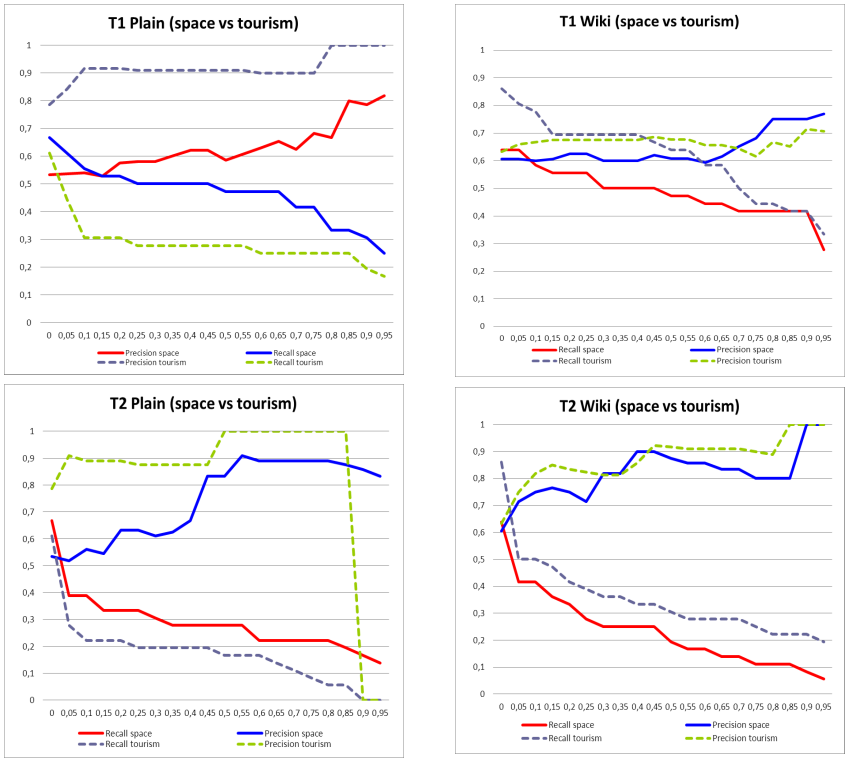


Figure 8 Influence of domain ontologies

5.5 Conclusions

In this chapter, an evaluation of the more relevant aspects of the feature extraction algorithm has been presented. The evaluation on any unsupervised automatic domain-independent extraction process is a hard task. On one hand, the evaluation has been performed through the intervention of a human expert in a particular domain that is represented by the input domain ontology. On the other hand, the final feature extraction and annotation is slanted by the precision and recall of each method used during the whole process (i.e., the natural language parser, the named entity detection heuristics, the inaccuracy of Web statistics and the relatedness measures). Considering these restrictions we can

conclude that the final feature annotations are certainly usable, even reaching a 100% precision in some cases.

Furthermore, the influence of all the input parameters has been studied. As it has been stated, thresholds can be tuned to modify the behaviour of the algorithm in order to improve either the precision or the recall. The threshold for named entity filtering is adequate to drop some named entities and decrease the number of queries needed during the whole process. The threshold to choose the proper annotation for each named entity is more restrictive and has a double purpose: 1) measure the relatedness degree between the named entity and its subsumer and 2) contextualize the ontology annotation in the domain of the analysed entity. It is important to note that considering that the final goal of the method is to enable the application of data analysis methods (such as clustering) a high precision is desirable, even at the cost of a reduced recall and, for that reason, the selection thresholds can be tuned for a high precision establishing a more restrictive value. Concerning the analysis of plain text and semi-structured resources like Wikipedia, it has been noticed that the analysis of unstructured documents is a hard and expensive task and taking profit of semi-structure of Wikipedia we can reach similar and even better results but with a considerably lower computational cost. Finally, the influence of the input domain ontology has been analysed in order to prove that the approach works in different domains. So, it is important to use a domain ontology proper to model the main concepts related with the area of study in order to maximize the quality of results.

6 Conclusions and future work

Since the creation of the World Wide Web (referred as WWW) its size and structure have been in constant growth and development. Nowadays the Web is in its second version known as the Social Web or Web 2.0. Due to the exponential growth of the available contents of Internet a new global initiative of the WWW has been proposed during the last years. This new approach is known as Semantic Web or Web 3.0.

One of the basic pillars of the Semantic Web concept is the idea of having explicit semantic information on the Web pages that can be used by intelligent agents in order to solve complex problems of Information Retrieval and Question Answering and to semantically analyze and catalog the Web contents. This fact has been implied the creation of new algorithms and semantic data mining techniques which are able to exploit semantic information, but many of them suppose that all the Web contents were annotated in advance. The manual annotation of Web contents is a hard task and hence semantic annotations are not available yet at a global scale.

This work aimed to extract and pre-process data from any kind of Web resource (i.e., plain text and semi-structured documents) in order to generate the required semantically tagged input data for aforementioned semantic data mining techniques. To sum up, in this work it has been designed and implemented a method that is able to extract relevant features from a range of textual documents going from complete plain textual data to semi-structured.

Extraction from plain text documents is more expensive than analysing semi-structured resources. This is because in plain text documents the analysis starts from scratch. By contrast, semi-structured resources present some particularities which facilitate the annotation process; for example, Wikipedia categories provide potential subsumer concepts for a particular real entity.

In order to reach the goals of the project (i.e., 1)identify relevant features describing a particular entity and 2)associate them, if it is possible, to concepts contained in an input ontologies) several techniques and tools have been used: *natural language processing parsers* have been useful to analyse texts and detect named entities, *Hearst Patterns* has been used to discover potential subsumer

concepts of named entities, Web scale statistics complemented with co-occurrence measures have been calculated to score and filter potential named entities and to verify if the final semantic annotation of subsumer concepts is applicable.

Being unsupervised and domain independent, all the implemented methodology has been designed in a generic way making possible its application in different domains and without human supervision.

6.1 Conclusions

Considering the developed methodologies and the evaluated and obtained results, we can conclude that:

- The Web is a valid corpus from where to extract information and it is actually the biggest repository of information in the world and its high redundancy can represent a measure of its relevance
- Named entities describe in a less unambiguous way than general entities a real entity. For that reason, they can be considered as features about the aforesaid real entity when they have been linked with concepts from an ontology (i.e., they have been semantically annotated).
- Lexico-syntactic patterns have been widely used in Information Retrieval and they are useful in order to discover taxonomic relations between named entities and ontological concepts (i.e., to discover potential subsumer concepts).
- The evaluations performed for several real entities with different ontologies have shown promising results to extract relevant features of the real entities.

6.2 Contributions

The main contributions of this work are:

- A list of the requirements that an Information Extraction system for the Web should accomplish, and a state of the art of the different existing techniques classifying them in function of their automatism grade (i.e., supervised and unsupervised approaches).
- A review of the use of the Web as a knowledge repository and a presentation of the tools and techniques which can be used in order to exploit Web contents.
- An automatic unsupervised scalable domain independent feature extraction methodology based on domain ontologies which allows the extraction of

relevant data from different types of resources (i.e., structured and semi-structured).

- A method to detect named entities and a method to extract potential subsumer concepts for those named entities. Both extracted from the Web in an unsupervised way.
- An automatic disambiguation method for semantic disambiguation based on the context Web snippets and contexts from where named entities have been extracted.

6.3 Future work

As further work, several research lines are proposed:

- It is a priority to study the quality of final extracted features when using them in semantic data mining algorithm and evaluates its applicability in different domains in order to perform clusters of real entities.
- Other important research future line is to study how to reduce the number of queries (e.g., using only a subset of Hearst Patterns) to Web search engines, as they are the slowest part of the algorithm and introduce a dependency on external resources.
- It is also important to evaluate the behaviour and applicability of the proposed methodology in other domains and ontologies.
- Analyse other kind of semi-structured resources and compare the influence of them in different domains. For example, Web blogs and its tags, available Linked Open Data, XML files, etc.
- Other priority task is the collection of a large set of documents annotated by experts in order to evaluate the automatic annotation performed by our approach with those ones did by experts.

6.4 Publications

The results of this work have been presented in the following articles:

- **Title:** Ontology-Based Feature Extraction

  Abstract: Knowledge-based data mining and classification algorithms require of systems that are able to extract textual attributes contained in raw text documents, and map them to structured knowledge sources (e.g. ontologies) so that they can be semantically analysed. The system presented in this paper performs this tasks in an automatic way, relying on a predefined ontology which states the concepts in this the posterior data

analysis will be focused. As features, our system focuses on extracting relevant Named Entities from textual resources describing a particular entity. Those are evaluated by means of linguistic and Web-based co-occurrence analyses to map them to ontological concepts, thereby discovering relevant features of the object. The system has been preliminary tested with tourist destinations and Wikipedia textual resources, showing promising results.

Conference: Workshop on 4th Natural Language Processing and Ontology Engineering (NLPOE 2011) is an international conference in conjunction with the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology (WI-IAT 2011). The main goal of the WI-IAT'11 workshops is to stimulate and facilitate active exchange, interaction and comparison of approaches, methods and ideas related to specific topics, both theoretical and applied, in the general areas related to Web Intelligence and Intelligent Agent Technology. The workshops will provide an informal setting where participants will have the opportunity to discuss specific technical topics in an atmosphere that fosters the active exchange of ideas.

State: ACCEPTED

- **Title:** A methodology to discover semantic features from textual resources

  Abstract: Data analysis algorithms focused on processing textual data rely on the extraction of relevant features from text and the appropriate association to their formal semantics. In this paper, a method to assist this task, annotating extracted textual features with concepts from a background ontology is presented. The method is automatic and unsupervised and it has been designed in a generic way, so it can be applied to textual resources ranging from plain text to semi-structured resources (like Wikipedia articles). The system has been tested with tourist destinations and Wikipedia articles showing promising results..

  Conference: SMAP 2011 is the 6th International Workshop on Semantic media adaptation and personalization. The SMAP initiative was founded during the summer of 2006 in an effort to discuss the state of the art, recent advances and future perspectives for semantic media adaptation and personalization.

  State: SUBMITTED

7 References

Agirre, E., O. Ansa, et al. (2000). Enriching very large ontologies using the www. Proceedings of the Ontology Learning Workshop, ECAI.

Ahmad, K., M. Tariq, et al. (2003). Corpus-Based Thesaurus Construction for Image Retrieval in Specialist Domains. Advances in Information Retrieval. F. Sebastiani, Springer Berlin / Heidelberg. **2633:** 76-76.

Alfonseca, E. and S. Manandhar (2002). Improving an ontology refinement method with hyponymy patterns. 3rd International Conference on Language Resources and Evaluation, LREC 2002, Las Palmas, Spain.

Banko, M. and O. Etzioni (2008). The Tradeoffs Between Open and Traditional Relation Extraction. 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, ACL 2008, Columbus, Ohio, USA pp. 28-36.

Batet, M., A. Valls, et al. (2010). Semantic clustering using multiple ontologies. 13th International Conference on the Catalan Association for Artificial Intelligence pp. 207-216.

Baumgartner, R., S. Flesca, et al. (2001). Visual Web Information Extraction with Lixto. 27th International Conference on Very Large Data Bases, VLDB 2001, Roma, Italy, Morgan Kaufmann pp. 119-128.

Berland, M. and E. Charniak (1999). Finding parts in very large corpora. Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics. College Park, Maryland, Association for Computational Linguistics**:** 57-64.

Berners-Lee, T., J. Hendler, et al. (2001). "The Semantic Web - A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities." Scientific American **284**(5): 34-43.

Berners-Lee, T., J. Hendler, et al. (2001). "The Semantic Web,." Scientific American Magazine **284**(5): 34-43.

Bisson, G., C. N\'dellec, et al. (2000). Designing clustering methods for ontology building: The {M}o'{K} workbench. Proceedings of the First Workshop on Ontology Learning OL'2000, Berlin, Germany, August 25, 2000. S. Staab, A. Maedche, C. N. dellec and P. Wiemer-Hastings.

Brill, E. (2003). Processing Natural Language without Natural Language Processing. 4th International Conference on Computational Linguistics and Intelligent Text Processing and Computational Linguistics, CICLing 2003, Mexico City, Mexico, Springer Berlin / Heidelberg pp. 360-369.

Brill, E., J. Lin, et al. (2001). Data-intensive question answering. In Procs. of Text REtrieval Conference (TREC-10**:** 393--400.

Buitelaar, P., P. Cimiano, et al. (2008). "Ontology-based information extraction and integration from heterogeneous data sources." International Journal of Human-Computer Studies **66**(11): 759 - 788.

Buitelaar, P., D. Olejnik, et al. (2004). A Protégé plug-in for ontology extraction from text based on linguistic analysis. In Proceedings of the 1st European Semantic Web Symposium (ESWS**:** 31--44.

Bunescu R.(2003): Associative Anaphora Resolution: A Web-Based Approach. In: Proc. of the EACL-2003 Workshop on the Computational Treatment of Anaphora.

Cafarella, M., D. Downey, et al. (2005). KnowItNow: fast, scalable information extraction from the web. Human Language Technology Conference/Conference on Empirical Methods in Natural Language Processing, HLT/EMNLP 2005, Vancouver, Canada, Association for Computational Linguistics pp. 563 - 570.

Califf, M. E. and R. J. Mooney (2003). "Bottom-up relational learning

of pattern matching rules for information extraction." The Journal of Machine Learning Research **4**(2): 177-210.

Calvo, H. and A. Gelbukh (2003). Improving Prepositional Phrase Attachment Disambiguation Using the Web as Corpus. Progress in Pattern Recognition, Speech and Image Analysis. A. Sanfeliu and J. Ruiz-Shulcloper, Springer Berlin / Heidelberg. **2905:** 604-610.

Caraballo, S. A. (1999). Automatic construction of a hypernym-labeled noun hierarchy from text. Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics. College Park, Maryland, Association for Computational Linguistics**:** 120-126.

Cilibrasi, R. L. and P. M. B. Vitányi (2006). "The Google Similarity Distance." IEEE Transactions on Knowledge and Data Engineering **19**(3): 370-383.

Cimiano, P. (2006). Ontology Learning and Population from Text, Springer-Verlag.

Cimiano, P. (2006). Text Analysis and Ontologies. Summer School on Multimedia Semantics, Kallithea, Chalkidiki, Greece.

Cimiano, P., S. Handschuh, et al. (2004). Towards the self-annotating web. 13th international conference on World Wide Web, WWW 2004, New York, USA, ACM pp. 462 - 471.

Cimiano, P., S. Handschuh, et al. (2004). Towards the self-annotating web. 13th international conference on World Wide Web, New York, NY, USA, ACM pp. 462 - 471.

Cimiano, P., G. Ladwig, et al. (2005). Gimme' The Context: Context-driven Automatic Semantic Annotation with C-PANKOW. 14th international conference on World Wide Web, Chiba, Japan, ACM pp. 462 - 471.

Ciravegna, F., A. Dingli, et al. (2003). Y.: Integrating Information to Bootstrap Information Extraction from Web Sites. In: IJCAI'03 Workshop on Intelligent Information Integration.

Ciravegna, F., A. Dingli, et al. (2002). User-system cooperation in document annotation based on information extraction. 13th International Conference on Knowledge Engineering and

Knowledge Management. Ontologies and the Semantic Web, Sigüenza, Spain, Springer Berlin / Heidelberg pp. 122-137.

Cunningham, H., D. Maynard, et al. (2002). GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. 40th Anniversary Meeting of the Association for Computational Linguistics, ACL 2002, Philadelphia, US.

Church, K., W. Gale, et al. (1991). Using Statistics in Lexical Analysis. Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon. U. Zernik. Hillsdale, New Jersey, USA, Lawrence Erlbaum Associates**:** 115-164.

Daudé, J., L. Padró, et al. (2003). Validation and Tuning of Wordnet Mapping Techniques. In Proceedings of RANLP, Borovets.

Dias, G., C. Santos, et al. (2006). Automatic knowledge representation using a graph-based algorithm for language-independent lexical chaining. Proceedings of the Workshop on Information Extraction Beyond The Document. Sydney, Australia, Association for Computational Linguistics**:** 36-47.

Dill, S., N. Eiron, et al. (2003). "A case for automated large-scale semantic annotation." Web Semantics: Science, Services and Agents on the World Wide Web **1**(1): 115-132.

Ding, L., T. Finin, et al. (2004). Swoogle: A Search and Metadata Engine for the Semantic Web. Thirteenth ACM International Conference on Information and Knowledge Management, CIKM 2004, Washington, DC, USA, ACM Press pp. 652-659.

Downey, D., M. Broadhead, et al. (2007). Locating complex named entities in Web text. 20th International Joint Conference on Artificial Intelligence, IJCAI 2007, Hyderabad, India, AAAI pp. 2733-2739.

Embley, D. W., D. M. Campbell, et al. (1998). Ontology-Based Extraction and Structuring of Information from Data-Rich Unstructured Document. Seventh ACM International Conference on Information and Knowledge Management, CIKM 1998, Bethesda, Maryland, USA, ACM Press pp. 52-59.

Etzioni, O., M. Banko, et al. (2008). "Open information extraction

from the web." Communications of the ACM **51**(12): 68-74.
Etzioni, O., M. Cafarella, et al. (2004). Web-scale information extraction in knowitall: (preliminary results). Proceedings of the 13th international conference on World Wide Web. New York, NY, USA, ACM**:** 100-110.
Etzioni, O., M. Cafarella, et al. (2005). "Unsupervised Named-Entity Extraction from the Web: An Experimental Study." Artificial Intelligence **165**(1): 91–134.
Etzioni, O., M. Cafarella, et al. (2005). "Unsupervised named-entity extraction form the Web: An experimental study." Artificial Intelligence **165**: 91-134.
Feilmayr, C., S. Parzer, et al. (2009). "Ontology-Based Information Extraction from Tourism Websites." Journal of Information Technology **11**(3): 183-196.
Fellbaum, C., Ed. (1998). WordNet: An electronic lexical database. Massachusetts, USA, MIT Press.
Fensel, D., C. Bussler, et al. (2002). Semantic web application areas. In Proceedings of the 7th International Workshop on Applications of Natural Language to Information Systems (NLDB.
Fleischman, M. and E. Hovy (2002). Fine grained classification of named entities. Proceedings of the 19th international conference on Computational linguistics - Volume 1. Taipei, Taiwan, Association for Computational Linguistics**:** 1-7.
Flesca, S., G. Manco, et al. (2004). "Web wrapper induction: a brief survey." AI Communications **17**(2): 57-61.
Freitag, D. (1998). Toward General-Purpose Learning for Information Extraction. 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, COLING-ACL 1998, Montreal, Quebec, Canada, ACL / Morgan Kaufmann pp. 404-408.
Freitag, D. and A. McCallum (1999). Information extraction with HMMs and shrinkage. AAAI-99 Workshop on Machine Learning for Information Extraction, Orlando, Florida, USA, AAAI pp. 31-36.
Gaizauskas, R. and Y. Wilks (1998). "Information Extraction: Beyond

Document Retrieval." Computacional Linguistics and Chinese Language Processing **3**(2): 17-60.

Geleijnse, G., J. Korst, et al. (2006). Google-based Information Extraction. 6th Dutch-Belgian Information Retrieval Workshop, DIR 2006, Delft, The Netherlands pp. 39-46.

Gómez-Pérez, A., M. Fernández-López, et al. (2004). Ontological Engineering, Springer-Verlag.

Gruber, T. R. (1995). "Toward principles for the design of ontologies used for knowledge sharing." Int. J. Hum.-Comput. Stud. **43**(5-6): 907-928.

Guarino, N. (1998). Formal Ontology in Information Systems. 1st International Conference on Formal Ontology in Information Systems, FOIS 1998, Trento, Italy, IOS Press pp. 3-15.

Hahn, U. and K. Schnattinger (1998). Towards text knowledge engineering. Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence. Madison, Wisconsin, United States, American Association for Artificial Intelligence**:** 524-531.

Handschuh, S., S. Staab, et al. (2003). Leveraging Metadata Creation for the Semantic Web with CREAM. 26th Annual German Conference on AI, KI 2003, Hamburg, Germany, Springer Berlin / Heidelberg pp. 19-33.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. 14th conference on Computational linguistics - Volume 2, COLING 92, Nantes, France, Morgan Kaufmann Publishers pp. 539 - 545.

Hotho, A., A. Maedche, et al. (2002). "Ontology-based Text Document Clustering."

Jarmasz, M. and S. Szpakowicz (2003). Roget's Thesaurus and Semantic Similarity. Conference on Recent Advances in Natural Language Processing, RANLP 2003, Borovets, Bulgaria pp. 212-219.

Jiang, J. J. and D. W. Conrath (1997). Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. International Conference on Research in Computational Linguistics, ROCLING X, Taipei, Taiwan pp. 19-33.

Keller, F., M. Lapata, et al. (2002). Using the web to overcome data sparseness. Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10, Association for Computational Linguistics**:** 230-237.

Kiryakov, A., B. Popov, et al. (2004). "Semantic annotation, indexing, and retrieval." Journal of Web Semantics **2**(1): 49-79.

Kiyavitskaya, N., N. Zeni, et al. (2005). Semi-Automatic Semantic Annotations for Web Documents 2nd Italian Semantic Web Workshop on Semantic Web Applications and Perspectives, SWAP 2005, Trento, Italy, CEUR-WS pp. 210-225.

Koivunen, M.-R. (2005). Annotea and Semantic Web Supported Collaboration (invited talk). Workshop on End User Aspects of the Semantic Web at 2nd Annual European Semantic Web Conference, UserSWeb 05 Heraklion, Crete, CEUR Workshop Proceedings pp. 5-17.

Kwok, C., O. Etzioni, et al. (2001). "Scaling question answering to the web." ACM Trans. Inf. Syst. **19**(3): 242-262.

Lamparter, S., M. Ehrig, et al. (2004). Knowledge Extraction from Classification Schemas. the Int. Conf. on Ontologies, Databases and Applications of SEmantics (ODBASE), Springer pp. 618-636.

Leacock, C. and M. Chodorow (1998). Combining local context and WordNet similarity for word sense identification. WordNet: An electronic lexical database, MIT Press**:** 265-283.

Lee, J. H., M. H. Kim, et al. (1993). "Information Retrieval Based on Conceptual Distance in Is-A Hierarchies." Journal of Documentation **49**(2): 188-207.

Li, Z. and K. Ramani (2007). "Ontology-based design information extraction and retrieval." Journal of Artificial Intelligence for Engineering Design, Analysis and Manufacturing, **21**(2): 137-154.

Lin, D. (1998). Automatic retreival and clustering of similar words. 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, COLING-ACL 1998, Montreal, Quebec, Canada, ACL / Morgan Kaufmann pp. 768-774.

Maedche, A., G. Neumann, et al. (2003). Bootstrapping an Ontology-based Information Extraction System. Intelligent exploration of the web. P. S. Szczepaniak, J. Segovia, J. Kacprzyk and L. A. Zadeh, Physica-Verlag**:** 345 - 359.

Markert, K., N. Modjeska, et al. (2003). Using the web for nominal anaphora resolution. In EACL Workshop on the Computational Treatment of Anaphora.

Matuszek, C., M. Witbrock, et al. (2005). Searching for common sense: populating cyc from the web. Twentieth National Conference on Artificial Intelligence (AAAI-05) and the Seventeenth Innovative Applications of Artificial Intelligence Conference (IAAI-05), Pittsburgh, Pennsylvania, USA, AAAI Press.

McCallum, A. (2003). Efficiently Inducing Features of Conditional Random Fields. 19th Conference in Uncertainty in Artificial Intelligence, UAI 2003, Acapulco, Mexico, Morgan Kaufmann pp. 403-410.

McDowell, L. K. and M. Cafarella (2008). "Ontology-driven, unsupervised instance population " Web Semantics: Science, Services and Agents on the World Wide Web **6**(3): 218-236

Michelson, M. and C. A. Knoblock (2007). An Automatic Approach to Semantic Annotation of Unstructured, Ungrammatical Sources: A First Look. IJCAI-2007 Workshop on Analytics for Noisy Unstructured Text Data, Hyderabad, India pp. 123-130.

Mikheev, A. and S. Finch (1997). A workbench for finding structure in texts. Proceedings of the fifth conference on Applied natural language processing. Washington, DC, Association for Computational Linguistics**:** 372-379.

Navigli, R. and P. Velardi (2004). "Learning Domain Ontologies from Document Warehouses and Dedicated Web Sites." Comput. Linguist. **30**(2): 151-179.

Neches, R., R. Fikes, et al. (1991). "Enabling technology for knowledge sharing." AI Mag. **12**(3): 36-56.

Nedellec, C. and A. Nazarenko (2005). Ontology and Information Extraction: A Necessary Symbiosis. Ontology Learning from Text: Methods, Evaluation and Applications. P. Buitelaar, P.

Cimiano and B. Magnini. Amsterdam, The Netherlands, IOS Press. **123:** 3-14.

Niekrasz, J. and A. Gruenstein (2006). NOMOS: A SemanticWeb Software Framework for Annotation of Multimodal Corpora 5th International Conference on Language Resources and Evaluation, LREC 06, Genoa, Italy pp. 21-27.

Pasca, M. (2004). Acquisition of categorized named entities for web search. Proceedings of the thirteenth ACM international conference on Information and knowledge management. Washington, D.C., USA, ACM**:** 137-145.

Paşca, M. (2005). Finding Instance Names and Alternative Glosses on the Web: WordNet Reloaded. Computational Linguistics and Intelligent Text Processing. A. Gelbukh, Springer Berlin / Heidelberg. **3406:** 280-292.

Porter, M. F. (1997). An algorithm for suffix stripping. Readings in information retrieval, Morgan Kaufmann Publishers Inc.**:** 313-316.

Resnik, P. (1999). "Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language." Journal of Artificial Intelligence Research **11**: 95-130.

Richardson, R., A. F. Smeaton, et al. (1994). Using WordNet as a Knowledge Base for Measuring Semantic Similarity between Words**:** 179--192.

Rijsbergen, C. J. v., S. E. Robertson, et al. (1980). "New models in probabilistic information retrieval."

Roberts, A., R. Gaizauskas, et al. (2007). The CLEF corpus: semantic annotation of clinical text. AMIA 2007 Annual Symposium, Chicago, USA, American Medical Informatics Association pp. 625-629.

Rosso, P., M. Montes, et al. (2005). Two Web-based Approaches for Noun Sense Disambiguation. In: Int. Conf. on Comput. Linguistics and Intelligent Text Processing, CICLing-2005, Springer Verlag, LNCS (3406), Mexico D.F**:** 261--273.

Rozenfeld, B. and R. Feldman (2008). "Self-supervised relation extraction from the Web." Knowl. Inf. Syst. **17**(1): 17-33.

Sánchez, D. (2008). Domain Ontology Learning from the Web, VDM Verlag.

Sánchez, D., D. Isern, et al. (2010). "Content Annotation for the Semantic Web: an Automatic Web-based Approach." Knowl. Inf. Syst., **27**(3): 393-418.

Sanderson, M. and B. Croft (1999). Deriving concept hierarchies from text. Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. Berkeley, California, United States, ACM**:** 206-213.

Schroeter, R., J. Hunterd, et al. (2003). Vannotea - A Collaborative Video Indexing, Annotation and Discussion System for Broadband Networks. Knowledge Markup and Semantic Annotation Workshop, K-CAP 03, Sanibel, Florida, ACM pp. 9-26.

Skounakis, M., M. Craven, et al. (2003). Hierarchical hidden markov models for information extraction. 18th International Joint Conference on Artificial Intelligence, IJCAI 2003, Acapulco, Mexico, Morgan Kaufmann pp. 427-433.

Soderland, S. (1999). "Learning information extraction rules for semistructured and free text." Machine Learning **34**(1-3): 233-272.

Solorio, T., M. P\, et al. (2004). A language independent method for question classification. Proceedings of the 20th international conference on Computational Linguistics. Geneva, Switzerland, Association for Computational Linguistics**:** 1374.

Staab, S. and A. Maedche (2000). Ontology engineering beyond the modeling of concepts and relations. ECAI-2000 Workshop on Ontologies and Problem-Solving Methods, Berlin, Germany.

Stephen Jose, H., D. C. Jack, et al. (1993). Advances in Neural Information Processing Systems 5, [NIPS Conference]. NIPS, Denver, Colorado, USA, Morgan Kaufmann.

Stevenson, M. and R. Gaizauskas (2000). Using corpus-derived name lists for named entity recognition. Proceedings of the sixth conference on Applied natural language processing. Seattle, Washington, Association for Computational Linguistics**:** 290-295.

Studer, R., V. R. Benjamins, et al. (1998). "Knowledge Engineering: Principles and Methods." IEEE Transactions on Knowledge and Data Engineering **25**(1-2): 161-197.

Studer, R., V. R. Benjamins, et al. (1998). "Knowledge Engineering: Principles and Methods." Data & Knowledge Engineering **25**(1-2): 161-197.

Stumme, G., M. Ehrig, et al. (2003). The Karlsruhe View on Ontologies. Karlsruhe, Germany, Institute AIFB, Universität Karlsruhe.

Surowiecki, J. (2004). The wisdom of crowds : why the many are smarter than the few and how collective wisdom shapes business, economies, societies, and nations. New York, Doubleday.

Turney, P. D. (2001). Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL. 12th European Conference on Machine Learning, ECML 2001, Freiburg, Germany, Springer-Verlag pp. 491-502.

Uschold, M. and M. Gruninger (1996). "Ontologies: principles, methods and applications." Knowledge Engineering Review(11): 112--114.

van Hage, W. R., S. Katrenko, et al. (2005). A Method to Combine Linguistic Ontology-Mapping Techniques. 4th International Semantic Web Conference, ISWC 2005 Galway, Ireland pp. 732-744.

Velardi, P., R. Navigli, et al. (2005). Evaluation of {OntoLearn,} a Methodology for Automatic Learning of Domain Ontologies. In Buitelaar, P., O. Cimiano & B. Magnini (eds.). Ontology Learning from Text: Methods, Evaluation and Applications, {IOS} Press.

Vicient, C. (2009). Extracció basada en ontologies d'informació de destinacions turístiques a partir de la Wikipedia. Tarragona, Universitat Rovira i Virgili.

Volk, M. (2001). Exploiting the WWW as a corpus to resolve PP attachment ambiguities. Proceedings of the Corpus Linguistics Conference**:** 601--606.

Volk, M. (2002). Using the web as corpus for linguistic research.

ähendusepüüdja. Catcher of the Meaning. A Festschrift for Professor Haldur Õim. R. Pajusalu and T. Hennoste. Tartu, University of Tartu**:** 1-10.

Wu, Z. and M. Palmer (1994). Verb semantics and lexical selection. 32nd annual Meeting of the Association for Computational Linguistics, Las Cruces, New Mexico, Association for Computational Linguistics pp. 133 -138.

Xiao, L., D. Wissmann, et al. (2004). "Information Extraction from the Web: System and Techniques " Applied Intelligence **21**(2): 195-224.

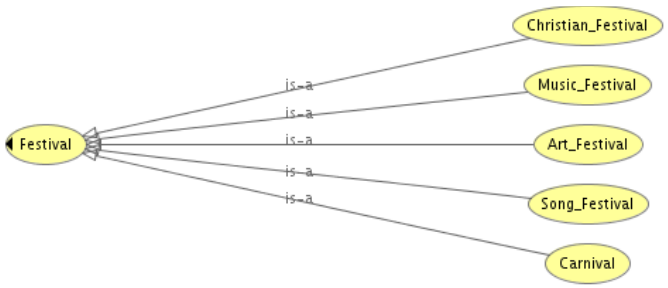
Yangarber, R., R. Grishman, et al. (2000). Unsupervised Discovery of Scenario-Level Patterns for Information Extraction. 18th International Conference on Computational Linguistics, COLING 2000, Saarbrücken, Germany, Morgan Kaufmann pp. 940-946.

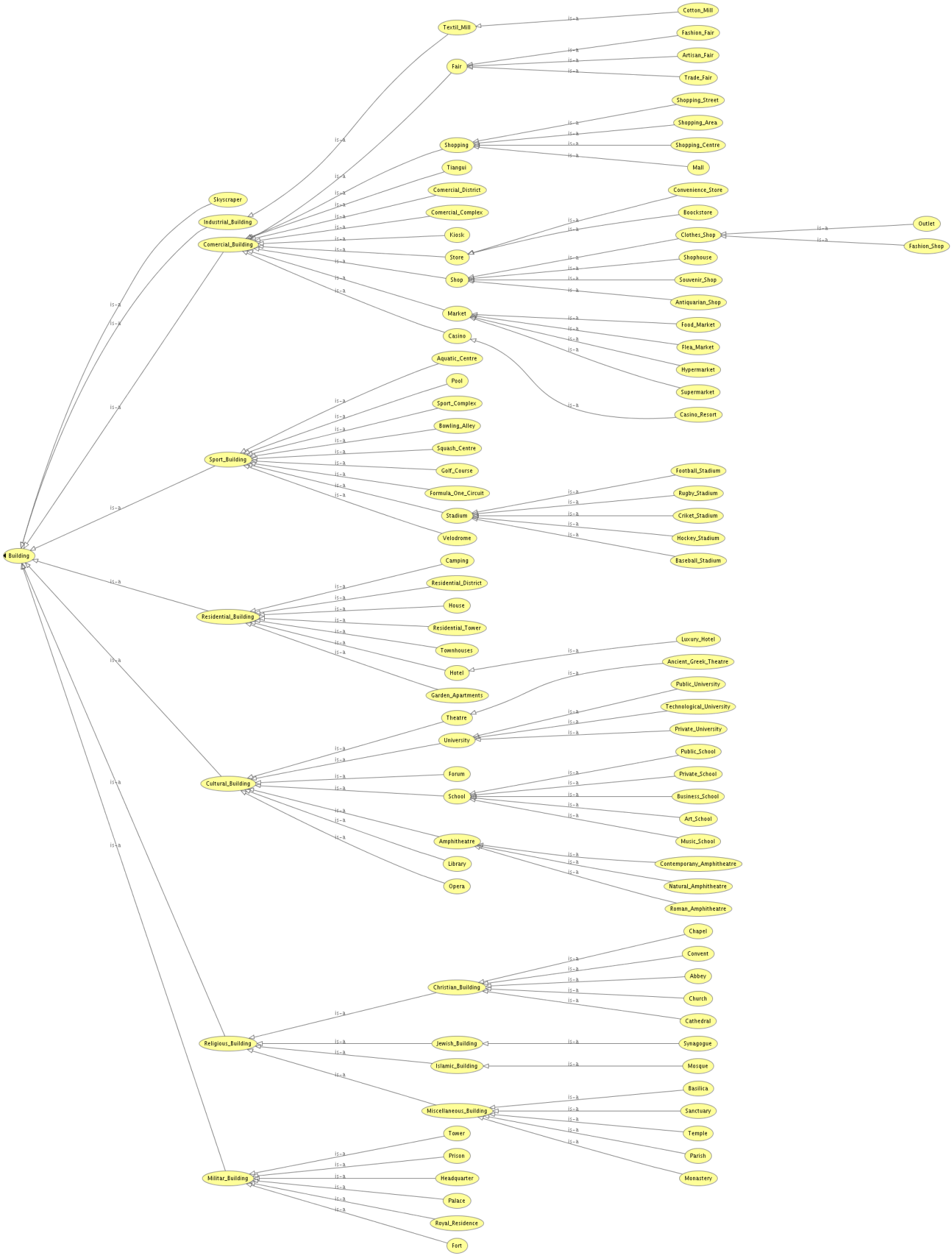
Yildiz, B. and S. Miksch (2007). Motivating ontology-driven information extraction. International Conference on Semantic Web and Digital Libraries, Bangalore, India, Indian Statistical Institute Platinum pp. 45–53.

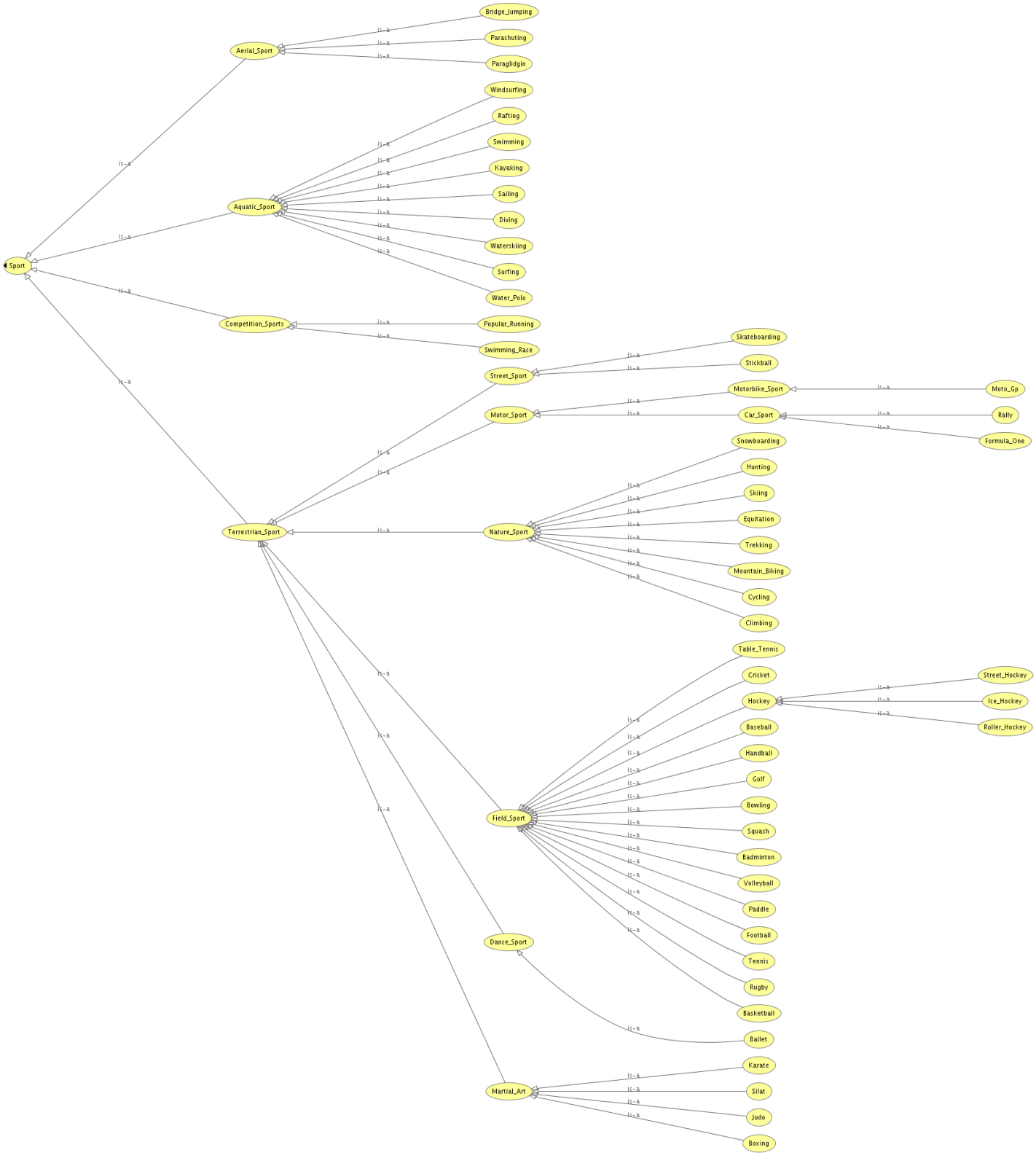
Annex I – TourismOWL

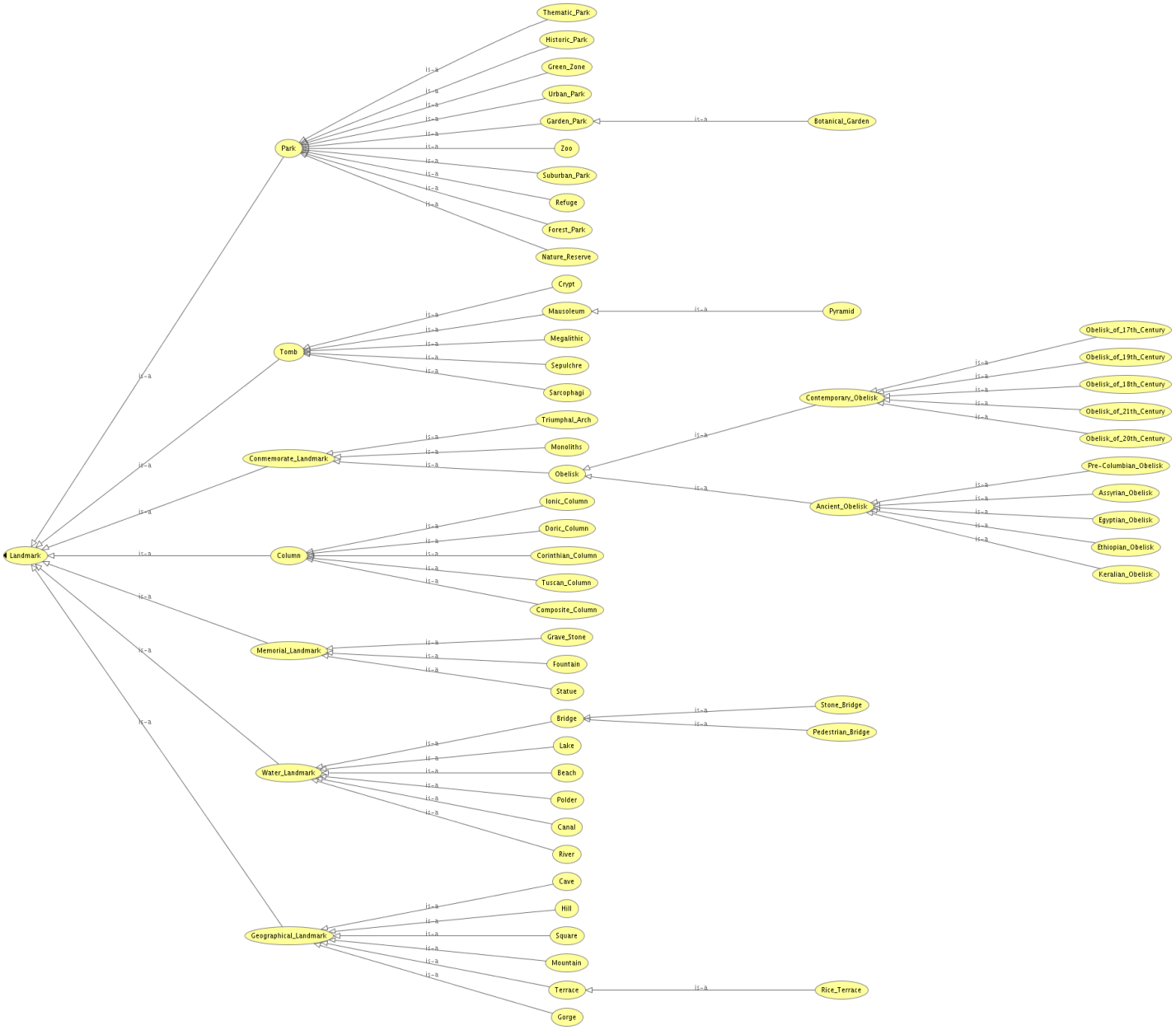
Following it is shown the taxonomy of TourismOWL.owl ontology. Each sub-tree depicts the main classes of the ontology and its hierarchy.


Contemporary_Art_Museum
Folk_Art_Museum
Art_Museum
Modern_Art_Museum
Art_Gallery
Industrial_Museum
Railway_Museum
Aviation_Museum
Maritime_Museum
Astronomy_Museum
Science_Museum
Computer_Museum
Technology_Museum
Physics_Museum
Children_Museum
Toy_Museum
Wax_Museum
Feminist_Museum
Specialized_Museum
Biographical_Museum
Virtual_Museum
Fishing_Museum
Museum
Egyptian_Museum
Music_Museum
Sex_Museum
Military_Museum
Natural_History_Museum
Woman_Museum
Open_Air_Museum
Erotic_Museum
Archeology_Museum
Ecomuseum
Mobile_Museum
Car_Museum
is-a


Christian_Festival
Music_Festival
Festival
Art_Festival
Song_Festival
Carnival
is-a

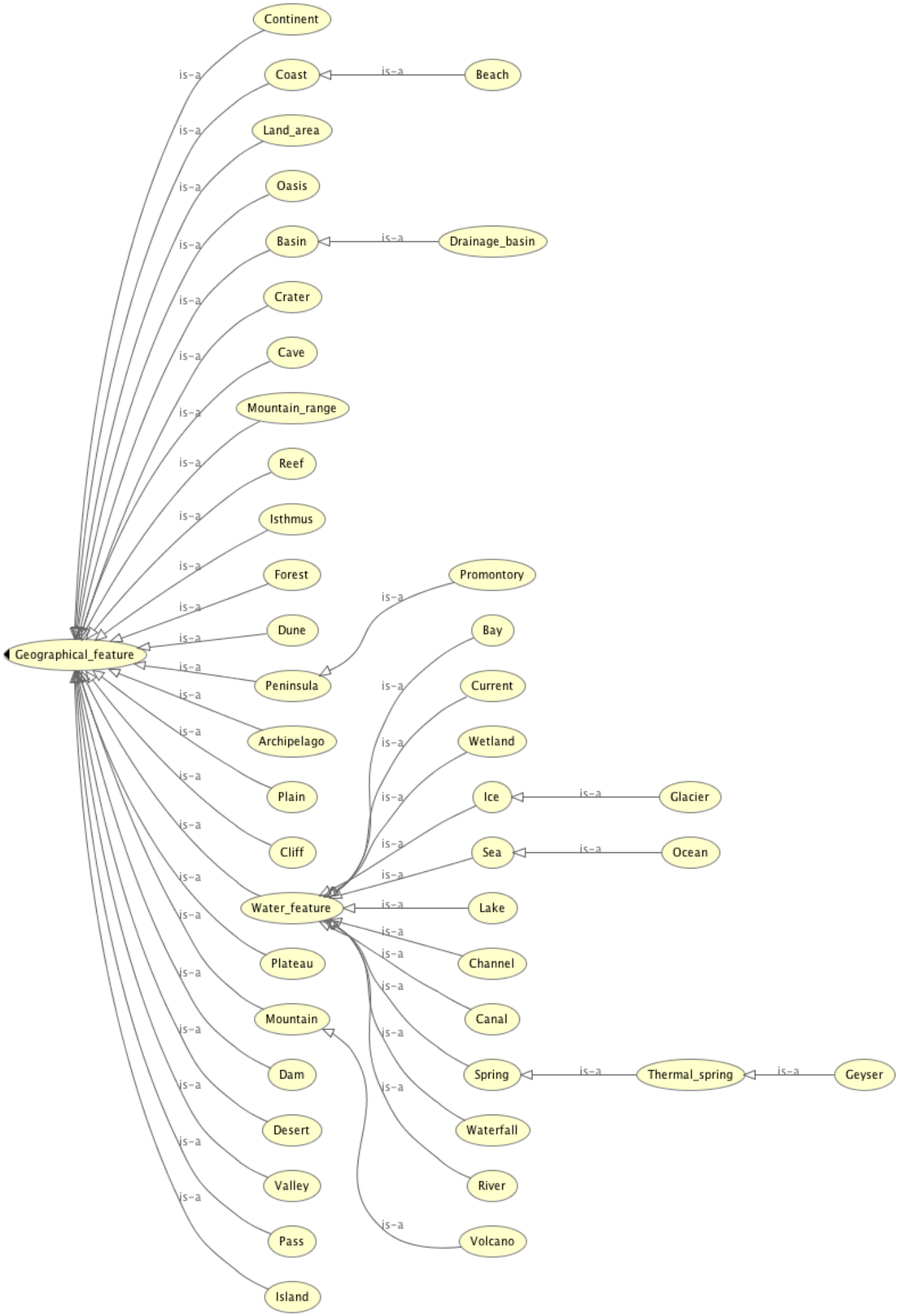

Building
Skyscraper
Industrial_Building
Comercial_Building
Sport_Building
Residential_Building
Cultural_Building
Religious_Building
Militar_Building
Textil_Mill
Cotton_Mill
Fair
Fashion_Fair
Artisan_Fair
Trade_Fair
Shopping
Shopping_Street
Shopping_Area
Shopping_Centre
Mall
Tiangui
Comercial_District
Comercial_Complex
Kiosk
Store
Convenience_Store
Boockstore
Clothes_Shop
Outlet
Fashion_Shop
Shop
Shophouse
Souvenir_Shop
Antiquarian_Shop
Market
Food_Market
Flea_Market
Hypermarket
Supermarket
Casino
Casino_Resort
Aquatic_Centre
Pool
Sport_Complex
Bowling_Alley
Squash_Centre
Golf_Course
Formula_One_Circuit
Stadium
Football_Stadium
Rugby_Stadium
Criket_Stadium
Hockey_Stadium
Baseball_Stadium
Velodrome
Camping
Residential_District
House
Residential_Tower
Townhouses
Hotel
Luxury_Hotel
Garden_Apartments
Theatre
Ancient_Greek_Theatre
University
Public_University
Technological_University
Private_University
Forum
School
Public_School
Private_School
Business_School
Art_School
Music_School
Amphitheatre
Contemporany_Amphitheatre
Natural_Amphitheatre
Roman_Amphitheatre
Library
Opera
Christian_Building
Chapel
Convent
Abbey
Church
Cathedral
Jewish_Building
Synagogue
Islamic_Building
Mosque
Miscellaneous_Building
Basilica
Sanctuary
Temple
Parish
Monastery
Tower
Prison
Headquarter
Palace
Royal_Residence
Fort


Sport
Aerial_Sport
Bridge_Jumping
Parachuting
Paraglidgin
Aquatic_Sport
Windsurfing
Rafting
Swimming
Kayaking
Sailing
Diving
Waterskiing
Surfing
Water_Polo
Competition_Sports
Popular_Running
Swimming_Race
Terrestrian_Sport
Street_Sport
Skateboarding
Stickball
Motor_Sport
Motorbike_Sport
Moto_Gp
Car_Sport
Rally
Formula_One
Nature_Sport
Snowboarding
Hunting
Skiing
Equitation
Trekking
Mountain_Biking
Cycling
Climbing
Field_Sport
Table_Tennis
Cricket
Hockey
Street_Hockey
Ice_Hockey
Roller_Hockey
Baseball
Handball
Golf
Bowling
Squash
Badminton
Volleyball
Paddle
Football
Tennis
Rugby
Basketball
Dance_Sport
Ballet
Martial_Art
Karate
Silat
Judo
Boxing
is-a

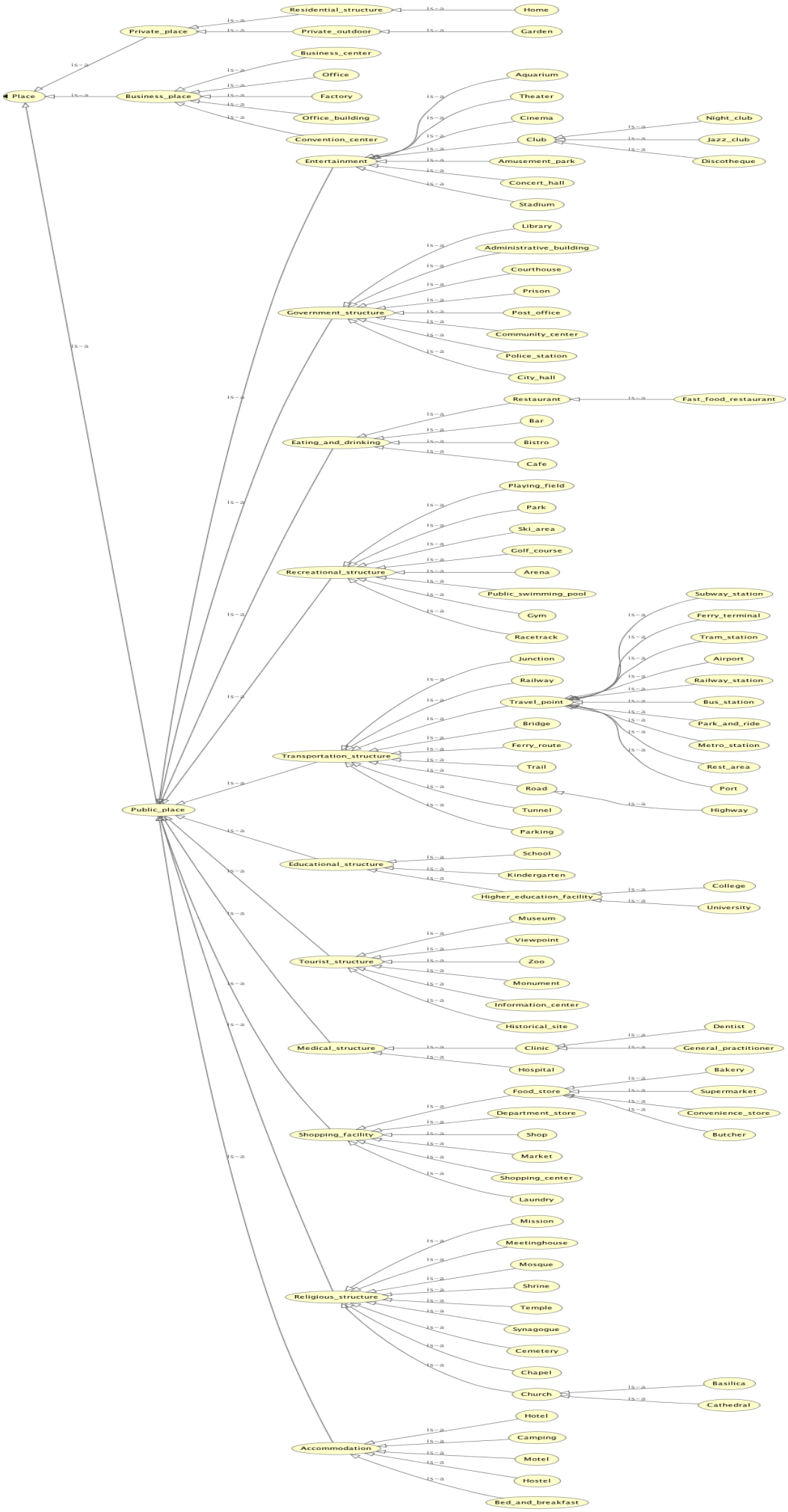

Landmark
is-a
Park
Thematic_Park
Historic_Park
Green_Zone
Urban_Park
Garden_Park
Botanical_Garden
Zoo
Suburban_Park
Refuge
Forest_Park
Nature_Reserve
Tomb
Crypt
Mausoleum
Pyramid
Megalithic
Sepulchre
Sarcophagi
Conmemorate_Landmark
Triumphal_Arch
Monoliths
Obelisk
Contemporary_Obelisk
Obelisk_of_17th_Century
Obelisk_of_19th_Century
Obelisk_of_18th_Century
Obelisk_of_21th_Century
Obelisk_of_20th_Century
Ancient_Obelisk
Pre-Columbian_Obelisk
Assyrian_Obelisk
Egyptian_Obelisk
Ethiopian_Obelisk
Keralian_Obelisk
Column
Ionic_Column
Doric_Column
Corinthian_Column
Tuscan_Column
Composite_Column
Memorial_Landmark
Grave_Stone
Fountain
Statue
Water_Landmark
Bridge
Stone_Bridge
Pedestrian_Bridge
Lake
Beach
Polder
Canal
River
Geographical_Landmark
Cave
Hill
Square
Mountain
Terrace
Rice_Terrace
Gorge

Annex II – Space.owl

Following it is shown the taxonomy of Space.owl ontology. Each sub-tree depicts the main classes of the ontology and its hierarchy.


Continent
Coast
Beach
Land_area
Oasis
Basin
Drainage_basin
Crater
Cave
Mountain_range
Reef
Isthmus
Forest
Promontory
Dune
Bay
Geographical_feature
Peninsula
Current
Archipelago
Wetland
Plain
Ice
Glacier
Cliff
Sea
Ocean
Water_feature
Lake
Plateau
Channel
Mountain
Canal
Dam
Spring
Thermal_spring
Geyser
Desert
Waterfall
Valley
River
Pass
Volcano
Island
is-a


Prefecture
Zone
Department
Street
Locality
District
Province
Area
Canton
Parish
Autonomous_Community
County
Subdivision
Republic
State
Monarchy
Emirate
Country
Geopolitical_entity
Hamlet
Capital
City
Village
Megalopolis
Town
Metropolis
Settlement
is-a


Place
Private_place
Residential_structure
Home
Private_outdoor
Garden
Business_place
Business_center
Office
Factory
Office_building
Convention_center
Public_place
Entertainment
Aquarium
Theater
Cinema
Club
Night_club
Jazz_club
Discotheque
Amusement_park
Concert_hall
Stadium
Government_structure
Library
Administrative_building
Courthouse
Prison
Post_office
Community_center
Police_station
City_hall
Eating_and_drinking
Restaurant
Fast_food_restaurant
Bar
Bistro
Cafe
Recreational_structure
Playing_field
Park
Ski_area
Golf_course
Arena
Public_swimming_pool
Gym
Racetrack
Transportation_structure
Junction
Railway
Travel_point
Subway_station
Ferry_terminal
Tram_station
Airport
Railway_station
Bus_station
Park_and_ride
Metro_station
Rest_area
Port
Bridge
Ferry_route
Trail
Road
Highway
Tunnel
Parking
Educational_structure
School
Kindergarten
Higher_education_facility
College
University
Tourist_structure
Museum
Viewpoint
Zoo
Monument
Information_center
Historical_site
Medical_structure
Clinic
Dentist
General_practitioner
Hospital
Shopping_facility
Food_store
Bakery
Supermarket
Convenience_store
Butcher
Department_store
Shop
Market
Shopping_center
Laundry
Religious_structure
Mission
Meetinghouse
Mosque
Shrine
Temple
Synagogue
Cemetery
Chapel
Church
Basilica
Cathedral
Accommodation
Hotel
Camping
Motel
Hostel
Bed_and_breakfast
is-a

Annex III – Contribution 1

Title: Ontology-Based Feature Extraction

Conference: Workshop on 4th Natural Language Processing and Ontology Engineering (NLPOE 2011) is an international conference in conjunction with the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology (WI-IAT 2011). The main goal of the WI-IAT'11 workshops is to stimulate and facilitate active exchange, interaction and comparison of approaches, methods and ideas related to specific topics, both theoretical and applied, in the general areas related to Web Intelligence and Intelligent Agent Technology. The workshops will provide an informal setting where participants will have the opportunity to discuss specific technical topics in an atmosphere that fosters the active exchange of ideas.

Ontology-Based Feature Extraction

Carlos Vicient, David Sánchez, Antonio Moreno
Universitat Rovira i Virgili
Departament d'Enginyeria Informàtica i Matemàtiques
Intelligent Technologies for Advanced Knowledge Acquisition Research Group (ITAKA)
Av Països Catalans, 26. 43007 Tarragona, Catalonia (Spain)
{carlos.vicient, david.sanchez, antonio.moreno}@urv.cat

***Abstract*—Knowledge-based data mining and classification algorithms require of systems that are able to extract textual attributes contained in raw text documents, and map them to structured knowledge sources (e.g. ontologies) so that they can be semantically analyzed. The system presented in this paper performs this tasks in an automatic way, relying on a predefined ontology which states the concepts in this the posterior data analysis will be focused. As features, our system focuses on extracting relevant Named Entities from textual resources describing a particular entity. Those are evaluated by means of linguistic and Web-based co-occurrence analyses to map them to ontological concepts, thereby discovering relevant features of the object. The system has been preliminary tested with tourist destinations and Wikipedia textual resources, showing promising results.**

***Keywords*-Ontologies, Information Extraction, Linguistic Patterns, Web-based statistics;**

I. Introduction

The Information Society provides users access to large amounts of electronic resources, most of which are represented in textual data. Due to the interest in automated analysis of all this information and thanks to global initiatives such as the Semantic Web [1], which aims to bring semantics to Web content, in recent years, knowledge-based data mining and classification algorithms have been proposed [2]. These methods rely on predefined knowledge (such as ontologies[3]) to semantically interpret textual data and extract more accurate conclusions from their analyses. They are typically applied over structured textual attributes which correspond to features of the analysed entities. In these cases, attribute labels (i.e., words or noun phrases) are interpreted by mapping them to concepts and analysing the background knowledge structure to which these concepts belong. However, these methods are rarely able to deal with raw text, from which relevant features should be extracted and matched to ontological entities before the data analysis.

The work presented in this paper aims to ease the application of semantically-grounded data-mining algorithms on textual data. Starting from a textual source describing an entity (e.g., a Web site about a tourist destination), we present a system that is able to extract relevant textual attributes describing features of the analysed entity, and annotate, if it is possible, these features to concepts in a background ontology. Ideally, this ontology should model the knowledge domain in which the posterior data analysis will be focused (e.g. touristic points of interest).

To discover the relevant features of an object, we focus on the extraction and selection of *Named Entities (NEs)* found in the text. We assume that NEs describe, in a way less ambiguous than general words, relevant features of the analysed entity. A relevance-based analysis based on Web co-occurrence statistics is performed in order to select which of the NEs are the most related to (i.e., identify better) the analysed entity. Afterwards, the selected NEs are matched to the ontological concepts to which they could be considered as instances. In this manner the extracted features are presented in an annotated fashion, easing the posterior application of semantically-grounded data analyses. The whole process is unsupervised and automatic; thus, it is a scalable solution that can be applied regardless of the type of entities or the knowledge domain.

The rest of the paper is organised as follows. Section II introduces our methodology, which is composed of two different parts: Named Entities detection (Section II-A) and semantic matching (Section II-B). Section III presents some preliminary results when analysing tourist destination descriptions. Some related works are commented in section IV. The paper finishes with the conclusions and some lines of future work.

II. Methodology

The following subsections describe the two steps of the feature extraction process: the detection of Named Entities and the semantic matching between them and the input ontology concepts.

A. Named Entities detection and filtering

In several approaches in the field of NE detection (e.g., [4]), it has been proposed to use a thesaurus as background knowledge (i.e., if a word does not appear in a dictionary, it is considered as a NE). The main problem of this kind of approaches is that misspelled words are wrongly considered NEs, whereas correct NEs composed by a set of common words are rejected. Other approaches [5] rely on training examples to detect predefined types of NEs (such as names of persons or locations) but they have a low recall when dealing with unbounded NEs.

On the contrary, our methodology is unsupervised and starts by parsing a Web document, which is supposed to

describe a particular real world entity, from now on *Analized Entity* (*AE*). Then a linguistic analysis consisting on Sentence Detection, Tokenizer, Tagging and Chunking is applied. As a result, Noun Phrases (NP) are detected. Those which fulfill a *capitalization heuristic* (i.e., NPs containing at least one word that begins with a capital letter) are selected as a Potential Named Entities (PNEs).

Some of the PNE describes the main features of AE; however, the rest of the elements may introduce noise in the posterior analysis because they are not directly related to the analysed entity (they just happen to appear in the Web page describing the entity but are not part of its basic distinguishing characteristics). Thus, it is necessary to have a way of separating the relevant NEs from the irrelevant ones (NE filtering). To do that, we use a Web-based co-occurrence measure that tries to assess the degree of relationship between AE and each PNE. In fact, it has been stated that the amount and heterogeneity of information in the Web is so high that it can be assumed to approximate the real distribution of information in the world ([6]). Concretely, a version of the *Pointwise Mutual Information* (PMI) relatedness measure adapted to the Web is computed. The score (Eq. 1) computes the probability of the co-occurrence of two terms from the Web hit count provided by a search engine when querying each of the terms separately.

$$NE_{score}(PNE_i, AE) = \frac{hits(PNE_i \ and \ AE)}{hits(PNE_i)} \qquad (1)$$

This score, as presented by Turney[7], statistically assesses the relation between two words (*a*, *b*) as the conditional probability of *a* and *b* co-occurring within the text. In Eq. 1, concept probabilities are approximated by Web hit counts provided by a Web search engine. Concretely, hits(PNE_i and AE) is the probability that PNE_i and AE co-occur in a Web page. Finally, the NEs that have a score exceeding an empirically determined threshold are considered as relevant, whereas the rest are removed. The value of the threshold will determine a compromise between the precision and the recall of the system.

B. Semantic Matching

The aim of this step is to match the NEs with the appropriate ontology classes. One way to assess the relationship between two terms (which, in our case, would be a NE and an ontology class) is to use a general thesaurus like Wordnet to compute a similarity measure based on the number of semantic links among the queried terms ([8]). However, those measures are hampered by WordNet's limited coverage of NEs and, in consequence, it is usually not possible to compute the similarity between a NE and an ontological class in this way. There are other approaches which try to discover automatically taxonomic relationships ([9]), but they require a considerable amount of background documents and linguistic parsing. Finally, another possibility is to compute the co-occurrence between each NE and each ontological class using Web-scale statistics as the relatedness measure ([7]), but this solution is not scalable because of the huge amount of required queries ([10]). We will use this last technique, but introducing a previous step that reduces the number of queries to be performed.

TABLE I
PATTERNS USED TO RETRIEVE POTENTIAL SUBSUMER CONCEPTS

Pattern structure	**Query**	**Example**
CONCEPT such as NE	"such as Barcelona"	**cities** such as Barcelona
such CONCEPT as NE	"such * as Spain"	such **countries** as Spain
NE and other CONCEPT	"Ebre and other"	Ebre and other **rivers**
NE or other CONCEPT	"The Sagrada Familia or other"	The Sagrada Familia or other **monuments**
CONCEPT especially NE	"especially Tarragona"	**World Heritage Sites** especially Tarragona
CONCEPT including NE	"including London"	**capital cities** including London

In our approach the semantic matching is divided in two parts: the discovery of potential subsumer concepts and their matching with the ontology classes.

1) Discovering potential subsumer concepts: To minimize the number of queries between NEs and ontology class to be performed, we propose to automatically discover ontology classes that are potentially good candidates for the matching process. If the number of candidates is small, it will feasible to use Web-scale statistics to compute the relatedness between them and each NE. It may be noticed that the problem is finding a bridge between the instance level (i.e., a NE) and the conceptual level (i.e. an ontology concept for which the NE is an instance). Semantically, NEs and concepts are related by means of *taxonomic relationships*. So, the way to go from the instance level to the conceptual level is by discovering taxonomic relationships.

We use the standard *Hearst's taxonomic linguistic patterns*, which have proved their effectiveness to discover *hyponym/hypernim* relationships [11]. We exploit the Web as the corpus from which to extract the semantic evidences of the appearances of the patterns. The system constructs a Web query for each NE and for each pattern. Each query is sent to a Web search engine, which returns as a result a set of Web snippets. Finally, all these snippets are analysed in order to extract a list of *potential subsumer concepts* (i.e., expressions that denote concepts of which the NE may be considered an instance). Table I summarizes the linguistic patterns that have been used (CONCEPT represents the retrieved potential subsumer concept and NE the studied Named Entity).

2) Ontology matching: The last step of the methodology aims to find a correspondence between the potential subsumers of each NE and the classes of an ontology. We use an input ontology in order to drive the mapping process and to indicate what kind of features are relevant in a particular domain. We will distinguish between two types of matching: *Direct Matching* and *Semantic Matching*. Moreover, there are situations in which there is evidence that a certain NE is related to several ontological classes. In this case, Web-based

statistical measures will be applied again in order to choose the most representative one (*Class Selection*). These three steps are explained in the following paragraphs.

a) Direct Matching: The system tries to find a direct match between the potential subsumers of a NE and the ontology classes. First, it extracts of all the classes contained in the domain ontology. Then, for each Named Entity NE_i, all its potential subsumer concepts (SC_i) are compared against each ontology class in order to discover the most similar ontological classes (SOC_i), i.e., classes whose name matches the subsumer itself or a subset of it (e.g., if one of the potential subsumers is "Gothic cathedral"', it would match an ontology class called "Cathedral"). A stemming algorithm is applied to both $SC_i s$ and ontology classes to discover terms that have the same root (e.g., "city" and "cities"). If one (or several) ontology classes match with the potential subsumers, they are included in SOC_i as candidates for the final annotation of NE_i. Even though, in many cases, the subsumers may not appear as ontology classes with exactly the same name, and potentially good candidates for annotation are not discovered.

b) Semantic Matching: The semantic matching step is performed when the direct matching has not produced any result. Its main goal is to increase the number of elements in SC_i, so that the direct matching can be tried again with a wider set of terms. The new potential subsumers are concepts semantically related to any of the initial subsumers (synonyms, hypernyms and hyponyms). As we are working at a conceptual level, WordNet has been used to obtain these related terms and to increase the set SC_i. The main problem of semantic matching is that many words are polysemous and, before extracting the related concepts from WordNet, we have to discover which is the WordNet sense that corresponds with the intended sense of the word in the discourse (i.e., a semantic disambiguation step must be performed).

For each element of SC_i of each NE_i, we look it up in WordNet. If it only has one definition (synset), the new subsumer candidates (synonyms, hypernyms and hyponyms) are retrieved. Otherwise, if the element of SC_i has more than one synset, it is necessary to choose the most suitable one. One possible solution is to use the context (i.e., the sentence from which NE_i was extracted) but, usually, this context is not enough to disambiguate the meaning. To minimize this problem, the Web is used again to extract new evidences of the relationship between NE_i and *AE*. A Web query containing *AE* and NE_i is performed, and a set of snippets are retrieved. Then, the system calculates the cosine distance between each snippet and all the synsets of the element of SC_i. The sense with a lower average value is finally selected.

c) Class Selection: When more than one ontology class has been proposed as annotation for a certain NE_i, the final step chooses the most appropriate one. The selection is based on the relatedness between the Named Entity and each element of SOC_i, assessed again with the Web-based version of PMI introduced in section II-A. The one with the highest score is selected. However, it must be noted that the elements of SOC_i can also be polysemous, and can be referring to different concepts depending on the context. So, in Eq. 2, the analysed entity *AE* has been introduced to contextualize the relationship of each element j of SOC_i with NE_i.

$$SOC_{score}(SOC_{ij}, NE_i, AE) = \frac{hits(AE\&NE_i\&SOC_{ij})}{hits(AE\&SOC_{ij})} \quad (2)$$

The score (E.q. 2) computes the probability of the co-occurrence of the named entity NE_i and each ontology class proposed for annotation SOC_{ij} from the Web hit count provided by a search engine when querying these two terms (contextualized with *AE*).

TABLE II
EVALUATION RESULTS

Measure	Detection	Filtering	Matching
Recall	60%	76%	25%
Precision	83%	71%	50%

III. PRELIMINARY RESULTS

In this section, we describe a preliminary test of the proposed system. The precision and recall of the NE extraction, filtering and ontology matching processes have been computed.

We picked up as *AE* the Lisburn[1] Wikipedia article with the *TourismOWL*[2] ontology. The content of of the Lisburn has been analysed as a plain text, and compared against the manually tagged items found in the Wikipedia article in order to compute the recall. We consider that tagged items found in Wikipedia articles are commonly referred to NEs. Moreover, an expert has selected which of those NE are strongly related (ENE) with the *AE*, taking into account the concepts contained in the domain ontology. Finally, for each NE from the expert NE list (ENE) he proposes a set of possible annotations which will be considered as AE's features.

First, it is evaluated the quality of the extracted NEs using the NLP package and the capitalization heuristic. Recall is computed by comparing them with those in the Wikipedia article (WNE). To calculate the precision, all the NEs extracted have been marked as either *Good NE* or *Bad NE* from an expert viewpoint. Next, it is estimated how many NEs have been dropped by the filter and how many of them should not had been rejected. In this step the main goal is delete all the NEs which are more general than the AE (e.g Ireland is more general than Lisburn), and all misspelled words. These are compared against those selected by the expert (ENE). The next test involves the evaluation of the ontology matching. To calculate the recall of annotation it is only taken into account the ENE list and its annotations. Then, each proposed annotation by the system is compared against the expert annotations. To calculate the precision all the annotations are manually mark, like in the first test, as either *Good Annotation* or *Bad Annotation*.

Table II shows the result of the evaluations.

Preliminary results show a reasonably good precision and recall for the NE extraction process, showing that most of the

[1]http://en.wikipedia.org/wiki/Lisburn
[2]http://deim.urv.cat/ itaka/TourismOWLv1.1.owl

relevant of the features of the evaluate AE could be detected. Ontology matching, on the other hand, shows lower values, as the matching recall depends on the coverage of the input ontology. In general, this is the most complex step, requiring proper extraction of candidates, disambiguation and matching.

IV. Related Work

Motivated by global initiatives such as the Semantic Web, several methods have been proposed to detect and annotate relevant entities from electronic resources (and, in particular, Web pages). On the one hand, there exist manual approaches providing tools to assist the user in the annotation process (such as Annotea [12]). On the other hand, some authors have tried to automate some of the stages of the annotation process to overcome the bottleneck of manual solutions. Melita [13] is based on user-defined rules and pre-defined annotations, which are employed to propose new annotations. Supervised approaches such as this one are difficult to apply, due to the effort required to compile a large and representative training set. Other systems like KnowItAll [14] rely on the redundancy of the Web to perform a bootstrapped information extraction process. The confirmation of the correctness of the obtained information is repeatedly requested to the user in order to re-execute the process with the support of the information obtained in the previous iteration.

Completely automatic and unsupervised annotation systems are rare. SemTag [15] performs automated semantic tagging based on the Seeker platform for text analysis. It has been able to tag a large number of pages with the terms included in a domain ontology named TAP. This ontology contains lexical and taxonomic information about areas like music, movies, sports and health, and SemTag detects the occurrence of entities related to these issues in Web pages. It disambiguates the retrieved terms by using neighbour tokens and corpus statistics, picking the best label for a token. From the applicability point-of-view, Pankow [16] is the most promising system. It uses a range of well-studied syntactic patterns to mark-up candidate phrases in Web pages. Its context driven version, C-Pankow [10], improves its computational efficiency by reducing the number of queries to the search engine. However, the final association between text entities and the classes of an input domain ontology is not addressed.

V. Conclusion

By using unsupervised information extraction and semantic annotation techniques, the proposed system is able to detect items that are relevant to describe an associated entity, matching them to their formal semantics, as modeled in an input ontology. System's results are useful to knowledge-based data mining algorithms which can exploit them to semantically classify entities according to their textual descriptions.

We are currently working in complementing some stages of the process with semi-structured information obtained from sources like Wikipedia (e.g., tagged entities, categories, etc.). In this manner, the accuracy of the results could be improved if some additional information can be extracted for the evaluated term. Moreover, we plan to perform extensive evaluations of the results, in order to test the influence of the different statistics, patterns and threshold used during the analysis.

Acknowledgements

This work has been partially supported by the Universitat Rovira i Virgili (predoctoral grant of C.Vicient, 2010BRDI-06-06), the Spanish Ministry of Science and Innovation (DAMASK project, Data mining algorithms with semantic knowledge, TIN2009-11005) and the Spanish Government (PlanE, Spanish Economy and Employment Stimulation Plan).

References

[1] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web : a new form of web content that is meaningful to computers will unleash a revolution of new possibilities," *Scientific American*, May 2001.

[2] M. Batet, A. Valls, K. Gibert, and D. Sánchez, *Semantic Clustering Using Multiple Ontologies. - Artificial intelligence research and development.*, IOS-Press, Ed. Proceedings of the 13th International Conference of the Catalan Association for Artificial Intelligence, 2010.

[3] N. Guarino, *Formal Ontology in Information Systems: Proceedings of the 1st International Conference on Formal Ontology in Information Systems, FOIS 1998, Trento, Italy, pp.3-15, 1998.* IOS Press, 1998.

[4] S. L. Marc, M. Ehrig, and C. Tempich, "Knowledge extraction from classification schemas," in *Proc. of the Int. Conf. on Ontologies, Databases and Applications of SEmantics (ODBASE.* Springer, 2004, p. 29.

[5] E. F. Tjong Kim Sang, "Introduction to the conll-2002 shared task: language-independent named entity recognition," in *proc. of the 6th conference on Natural language learning - Volume 20*, ser. COLING-02. Stroudsburg, PA, USA: Association for Computational Linguistics, 2002, pp. 1–4.

[6] R. L. Cilibrasi and P. M. B. Vitanyi, "The google similarity distance," *IEEE Trans. on Knowl. and Data Eng.*, vol. 19, pp. 370–383, 2007.

[7] P. D. Turney, "Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL," in *EMCL '01: Proc. of the 12th European Conference on Machine Learning*. London, UK: Springer-Verlag, 2001, pp. 491–502.

[8] Z. Wu and M. Palmer, "Verbs semantics and lexical selection," in *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, ser. ACL '94. Stroudsburg, PA, USA: Association for Computational Linguistics, 1994, pp. 133–138.

[9] G. Bisson, C. Nédellec, and D. C. namero, "Designing clustering methods for ontology building: The mo'k workbench," in *Proceedings of the ECAI Ontology Learning Workshop*, 2000, pp. 13–19.

[10] P. Cimiano, G. Ladwig, and S. Staab, "Gimme' the context: context-driven automatic semantic annotation with c-pankow," in *Proceedings of the 14th international conference on World Wide Web*, ser. WWW '05. New York, NY, USA: ACM, 2005, pp. 332–341.

[11] M. A. Hearst, "Automatic acquisition of hyponyms from large text corpora," in *Proceedings of the 14th conference on Computational linguistics - Volume 2*, ser. COLING '92. Stroudsburg, PA, USA: Association for Computational Linguistics, 1992, pp. 539–545.

[12] M. R. Koivunen, "Annotea and semantic web supported collaboration," *Workshop on End User Aspects of the Semantic Web at 2nd Annual European Semantic Web Conference*, pp. pp. 5–17, 2005.

[13] F. Ciravegna, A. Dingli, D. Petrelli, and Y. Wilks, "User-system cooperation in document annotation based on information extraction," in *Proc. of the 13th International Conference on Knowledge Engineering and Knowledge Management, EKAW02*. Springer, 2002, pp. 122–137.

[14] O. Etzioni, M. Cafarella, D. Downey, A.-M. Popescu, T. Shaked, S. Soderland, D. S. Weld, and A. Yates, "Unsupervised named-entity extraction from the web: an experimental study," *Artif. Intell.*, vol. 165, pp. 91–134, June 2005.

[15] S. Dill, N. Eiron, D. Gibson, D. Gruhl, R. Guha, A. Jhingran, T. Kanungo, K. S. Mccurley, S. Rajagopalan, A. Tomkins, J. A. Tomlin, and J. Y. Zien, "A case for automated large scale semantic annotations," *Journal of Web Semantics*, vol. 1, pp. 115–132, 2003.

[16] P. Cimiano, S. Handschuh, and S. Staab, "Towards the self-annotating web," in *Proceedings of the 13th international conference on World Wide Web*, ser. WWW '04. New York, NY, USA: ACM, 2004, pp. 462–471.

Annex IV - Contribution 2

Title: A methodology to discover semantic features from textual resources

Conference: SMAP 2011 is the 6th International Workshop on Semantic media adaptation and personalization. The SMAP initiative was founded during the summer of 2006 in an effort to discuss the state of the art, recent advances and future perspectives for semantic media adaptation and personalization.

A methodology to discover semantic features from textual resources

Carlos Vicient, David Sánchez, Antonio Moreno
Departament d'Enginyeria Informàtica i Matemàtiques
Universitat Rovira i Virgili
Av. Països Catalans, 26. 43007 Tarragona, Spain
{carlos.vicient, david.sanchez, antonio.moreno}@urv.cat

Abstract— **Data analysis algorithms focused on processing textual data rely on the extraction of relevant features from text and the appropriate association to their formal semantics. In this paper, a method to assist this task, annotating extracted textual features with concepts from a background ontology is presented. The method is automatic and unsupervised and it has been designed in a generic way, so it can be applied to textual resources ranging from plain text to semi-structured resources (like Wikipedia articles). The system has been tested with tourist destinations and Wikipedia articles showing promising results.**

Keywords: Ontologies; Information Extraction; Feature discovery; Wikipedia.

I. INTRODUCTION

The success of the Information Society has brought access to vast amounts of textual resources. This has motivated researchers in developing data analysis methods focused on textual data [1]. Most of these approaches, however, rely on an adequate mapping between textual features found in textual documents and their corresponding semantics. Global initiatives such as the Semantic Web[2] pursuit this goal, bringing semantic content to Web content by means of ontology-based semantic annotations. However, nowadays, the amount of semantically annotated The Web is still low in comparison to plain textual resources, because most initiatives are based on manual annotations.

In this paper, we present a generic and automatic method that aims to detect relevant features from textual documents, associating them to the concepts of a background ontology. The method is able to deal with raw texts, but also with semi-structured resources like Wikipedia. In this last case, available annotations (i.e., Wikilinks and associated categories in the Wikipedia's folksonomy) are exploited to potentially improve the accuracy of the feature extraction and ontology mapping processes and also the method's performance.

The rest of the paper is organised as follows. Section II introduces our generic method, detailing its application to plain text documents and Wikipedia articles. Section III presents some results obtained by analysing city descriptions. Section IV discusses related works. The final section contains the conclusions and several lines of future research.

II. GENERAL METHOD

Fig 1 shows the main steps of our methodology. Its design is generic in the sense that some functions can be adapted according to the type of analysed resources. It receives as input a textual document (e.g. Wikipedia article) describing an object (e.g. Barcelona) and an ontology modelling the concepts in which the posterior data analysis should be focused (e.g. points of interest).

In order to discover relevant features of an object, we focus on the extraction and selection of Named Entities (referred as NEs) from the text. It is assumed that NEs describe, in a way less ambiguous than general words, relevant features of the analysed entity. A relevance-based analysis based on Web co-occurrence statistics is performed in order to select which of the NEs are the most related to (i.e., identify better) the analysed entity. Afterwards, the selected NEs are matched to the ontological concepts to which they could be considered as instances. In this manner the extracted features are presented in an annotated fashion, easing the posterior application of semantically-grounded data analyses.

This section details the basic steps of the generic algorithm. Afterwards, the differences when applied to different types of input documents (i.e., plain text or semi-structured sources) are explained.

1) Document parsing

The algorithm starts by parsing the input document (line 3), that describes a particular object, from now on Analysed Entity (AE). As a result, the clean text is extracted.

2) Named Entity extration and selection

In this step, a set of Potential Named Entities (PNE) is extracted from the AE (line 6). The specific extraction procedure depends of the type of input document (see section III). Only a subset of the members of PNE will be adequate to describe the main features of AE. Thus, it is necessary to have a way of separating the relevant NEs from the irrelevant ones (NE filtering, line 8). To do that, we use a Web-based co-occurrence measure that assesses the degree of relationship between AE and each PNE. It relies on the fact that the amount and heterogeneity of the information in the Web is so high that it can be assumed to approximate the real distribution of information in society [3]. Concretely, a version of the Pointwise Mutual Information (PMI) relatedness measure adapted to the Web is computed[4].

In NE_{score} (1), concept probabilities are approximated by Web hit counts provided by a Web search engine. Those PNEs that have a score exceeding a threshold (*NE_THRESHOLD*) are considered as relevant, whereas the rest are removed. The value of the threshold will determine a compromise between the precision and the recall of the system.

$$NE_{score}(PNE_i, AE) = \frac{hits(PNE_i \cap AE)}{hits(PNE_i)} \quad (1)$$

```
1  OntologyBasedExtraction(WebDocument wd, String AE,  DomainOntology do){
2     /* Document Parsing */
3     pd ← parse_document(wd)
4
5     /* Extraction and selection of Named Entities from Document */
6     PNE ← extract_potential_NEs(pd)
7     ∀ pne_i ∈ PNE {
8           if NE_Score(pne_i, AE) > NE_THRESHOLD{
9                 NE ← NE ∪ pne_i
10          }
11    }
12    /* Retrieving potential subsumer concepts for each NE*/
13    ∀ ne_i ∈ NE  {
14          SC ← extract_subsumer_concepts(ne_i)
15          ne_i ← add_subsumer_concepts_list(SC)
16    }
17
18    /* Annotating NEs with ontological classes */
19    OC ← extract_ontological_classes(do)
20    ∀ ne_i ∈ NE  {
21          /* Retrieving Subsumer Ontological Classes (i.e. potential
22          annotations) for each Subsumer Concept of each NE*/
23          SC ← get_subsumer_concepts_list(ne_i)
24          /* Applying direct matching */
25          SOC ← extract_direct_matching(OC, SC)
26          /* if no direct matching, Semantic matching is applied*/
27          if |SOC| == 0 {
28                SOC ← extract_semantic_matching(OC, SC)
29          }
30          /* if a similar ontological class is found, the most proper
31          Annotation is chosen and the annotation is performed */
32          if |SOC| > 0 {
33                SOC ← SOC_Score(SOC, ne_i, AE)
34                ac ← select_SOC_max_score(SOC, AC_TRESHOLD)
35                ne_i ← add_annotation(ac)
36          }
37    }
38    return NE
39 }
```

Figure 1. Ontology-based feature extraction method.

3) Semantic annotation

This step aims to annotate the selected NEs with classes in the background ontology, in those cases in which it is possible to do so.

In the semantic annotation field, several approaches have been proposed to tackle this task. One way to assess the relationship between two terms (which, in our case, would be a NE and an ontology class) is to use a general thesaurus like WordNet to compute a similarity measure based on the number of semantic links among the queried terms[5]. However, when dealing with NEs, we are hampered by their limited coverage in WordNet.

Other approaches assess the relatedness between NEs and ontological classes according to their amount of co-occurrences in the Web[6], but this solution is not scalable because of the huge amount of required queries to evaluate the cartesian product between the sets of NEs and ontological classes [7].

Being an unsupervised algorithm, we follow a similar approach relying on Web statistics, but introducing an intermediate step to reduce the number of required queries. In our approach the semantic annotation is divided in two parts: the discovery of potential subsumer concepts (line 14) and their matching with the ontology classes. The goal is to be able to find matchings between subsumer concepts corresponding to the extracted NEs and ontological classes requiring, in the worst case, a reduced amount of web queries to perform the final assessment.

So, the first task of semantic annotation consists in discovering potential subsumer concepts (SC) for each NE. SCs are abstractions of real entities collections, which share some common characteristics. For example, the SC of the real entity *The Sagrada Familia* or *St. Peter's Basilica* is *basilica*. Notice that a real entity may belong to different concepts such as *basilica* and *monument*. Another important aspect is that SCs can be represented by different equivalent terms. Consider, for instance, the real entity *Porsche*, whose subsumer concept could be *car*, *automobile*, *auto*, *motorcar* and *machine*. Finally, the abstraction can be performed at different levels. In the case of *the Sagrada Familia* its direct subsumer is *basilica* but higher subsumer concepts such as *roman building* and *religious building* can be considered.

Semantically, NEs and subsumer concepts are related by means of *taxonomic relationships*. So, the way to go from the instance level to the conceptual level is by discovering taxonomic relationships. E*xtract_subsumer_concepts* function (line 14) obtains a set of potential subsumer concepts for each NE. This process will also depend on the type of input document (see details in section III).

Then, the last step of the methodology aims to try to find a correspondence between the subsumers of each NE and the classes of an ontology, if it is possible. We distinguish between two types of matching: *Direct Matching* (line 25) and *Semantic Matching* (line 28). Moreover, there are situations in which there is evidence that a certain NE is related to several ontological classes. In this case, Web-based statistical measures will be applied again in order to choose the most representative one (*Class Selection,* line 33-34).

In *Direct Matching*, the system tries to find a direct match between the potential subsumers of a NE and the ontology classes. This phase begins with the extraction of all the classes contained in the domain ontology. Then, for each Named Entity NE_i, all its potential subsumer concepts (SC_i) are compared against each ontology class to discover the most similar ontological classes (SOC_i), i.e., classes whose name matches the subsumer itself or a subset of it (e.g., if one of the potential subsumers is "Gothic cathedral", it would match an ontology class called "Cathedral"). A stemming algorithm is applied to both SC_i and ontology classes in order to discover terms that have the same root (e.g., "city" and "cities"). If one (or several) ontology classes match with the potential subsumers, they are included in SOC_i as candidates for the final annotation of NE_i. This direct matching step is quite easy and computationally efficient; however, its main problem is that, in many cases, the subsumers do not appear as ontology classes with exactly

the same name, and potentially good candidates for annotation are not discovered.

The *Semantic Matching* step is performed when the direct matching has not produced any result. Its goal is to increase the number of elements in SC_i, so that the direct matching can be tried again with a wider set of terms. The new potential subsumers are concepts semantically related to any of the initial subsumers (synonyms, hypernyms and hyponyms). As we are working at a conceptual level, WordNet has been used to obtain these related terms and to increase the set SC_i. The main problem of semantic matching is that many words are polysemous and, before extracting the related concepts from WordNet, we have to discover which is the *synset* that corresponds with the intended sense of the word in the domain (i.e., a semantic disambiguation step must be performed).

In order to deal with sense disambiguation, we propose a Web-based approach combining the context from which a NE has been extracted, WordNet definitions and the cosine distance. For each element of SC_i of each NE_i, we look it up in WordNet. If it only has one definition (synset), the new subsumer candidates (synonyms, hypernyms and hyponyms) are retrieved. Otherwise, if the element of SC_i has more than one synset, it is necessary to choose the most suitable one (word sense disambiguation). One possible solution is to use the context (i.e., the sentence from which NE_i was extracted) but, usually, this context is not enough to disambiguate the meaning. To minimize this problem, the Web is used again to extract new evidences of the relationship between NE_i and AE. A Web query containing AE and NE_i is performed, and a number of snippets are retrieved. Then, the system calculates the cosine distance between each snippet and all the synsets of the element of SC_i. The synset with a lower average value is finally selected.

Finally, the *Class selection* step is needed when more than one ontology class has been proposed as annotation for a certain NE_i. The final step is to choose the most appropriate one. The selection is based on the relatedness between the NE and each element of SOC_i, assessed again with the Web-based version of PMI. However, it must be noted that the elements of SOC_i can also be polysemous, and can be referring to different concepts depending on the context. So, in (2), the analysed entity AE has been introduced to contextualize the relationship of each element $_j$ of SOC_i with NE_i.

$$SOC_{score}(SOC_{ij}, NE_i, AE) = \frac{hits(AE \cap NE_i \cap SOC_{ij})}{hits(AE \cap SOC_{ij})} \qquad (2)$$

This score (2) computes the probability of the co-occurrence of the named entity NE_i and each ontology class proposed for annotation SOC_{ij} from the Web hit count provided by a search engine when querying these two terms (contextualized with AE).

III. PLAIN VS. SEMI-STRUCTURED RESOURCES

In this section, it is explained how to apply generic algorithm in either unstructured web resources (text plain) and semi-structured (Wikipedia articles).

A. Applying algorithm to plain texts

The features extraction process from plain text documents is the most difficult task. The main problems regard the extraction of NEs (line 6) and the discovery of SCs (line 14). In the following the details of how these tasks are tackled for this kind of documents are given.

1) Named Entity extration

The main problem related with *NE extraction* from raw text is the fact that they are unstructured and unlimited by nature[8]. This implies that, in most cases, these NEs are not contained in classical repositories as WordNet due to its potential size.

Supervised approaches try to detect NEs by using a specific set of extraction rules learned from pre-tagged examples[9], or predefined knowledge bases such as lexicons and gazetteers[10]. However, the amount of effort required to assemble large tagged sets or lexicons binds the NE recognition to either a limited domain (e.g., medical imaging), or to a small set of predefined, broad categories of interest (e.g., persons, countries, organizations, products). This fact introduces compromises in the recall [11].

In unsupervised approaches like [12], it has been proposed to use a thesaurus as background knowledge (i.e., if a word does not appear in a dictionary, it is considered as a NE). Despite the fact that this approach is not limited by the size of the thesaurus, misspelled words are wrongly considered as NEs whereas correct NEs composed by a set of common words are rejected, providing inaccurate results.

Other approaches take into consideration the way in which NEs are presented in the specific language. Concretely, languages such as English distinguish proper names from other nouns through capitalization. This simple idea, combined with linguistic pattern analysis, as it has been applied by several authors [11, 13], provides good results without depending on manually annotated examples or specific categories.

Being unsupervised, domain-independent and lightweight, in this work, this last approach has been implemented in order to detect NEs. A linguistic parsing consisting on Sentence Detection, Tokenization, Tagging and Chunking is applied. Then, all Noun Phrases which contain at least one word starting with a capital letter are considered as a PNE.

2) Discovering subsumer concepts

As stated in section II, the discovery of subsumer concepts for NEs requires the analysis of taxonomical relationships. We use the standard *Hearst's taxonomic linguistic patterns*, which have proved their effectiveness to retrieve *hyponym/hypernim* relationships [14]. We exploit the Web as the corpus from which to extract appearances of the patterns[15]. The main reason of using the Web as the corpus is because of the fact that explicit linguistic patterns

are difficult to find in reduced corpora, that normally offer a relatively high precision but suffer from low recall.

The system constructs a Web query for each NE and for each pattern. Each query is sent to a Web search engine, which returns as a result a set of Web snippets. Finally, all these snippets are analysed in order to extract a list of *Potential Subsumer Concepts* (i.e., expressions that denote concepts of which the NE may be considered an instance).

Table I. summarizes the linguistic patterns that have been used (CONCEPT represents the retrieved potential subsumer concept and NE the Named Entity that is being studied).

TABLE I. PATTERNS USED TO RETRIEVE PSC

Hearst Pattern	Query	Example
CONCEPT such as NE	"such as Barcelona"	*cities* such as Barcelona
such CONCEPT as NE	"such * as Spain"	Such *countries* as Spain
NE and other CONCEPT	"Ebre and other"	Ebre and other *rivers*
NE or other CONCEPT	"The Sagrada Familia or other"	The Sagrada Familia or other *monuments*
CONCEPT especially NE	"especially Tarragona"	*World Heritage Sites* especially Tarragona
CONCEPT including NE	"including London"	*capital cities* including London

B. Applying the algorithm to Wikipedia articles

Wikipedia has some particularities which can be useful when extracting information. Specially, in this work we can take profit of *internal links* and *category links*. The first ones represent connections between terms that appear in a Wikipedia article and other articles, which are talking about the aforementioned terms. *Category links* group different articles in areas that are related in some way and give articles a kind of categorization. The first ones will be used to automatically extract PNEs, whereas the second ones will aid in the process of semantic annotation.

1) Named Entity extracion

In order to take profit of links structure, the terms that contain an internal link will be considered as potential named entities (PNE). The hypothesis is that internal links have been created by a big community of users and it can be assumed that the information they represent has been revised for enough readers to assume that it is correct and relevant.

The problems of PNEs extracted from internal links are that, on one hand, not all of them are directly related with the analysed entity (*AE*) and, on the other hand, only a subset of PNEs are really NEs.

To illustrate these problems, the following fragment of text extracted from Wikipedia is examined. *"Barcelona is the capital and the most populous city of Catalonia and the second largest city in Spain, after Madrid, with […]"*. In this text, there are four terms linked with other Wikipedia articles by means of internal links. Three of them are NE (Catalonia, Spain and Madrid) and they represent instances of things; the other one is a common noun that represents a concept capital. Due to these problems, the extracted PNEs are filtered by means of the NE score presented in section II.

2) Discovering subsumer concepts

In order to extract subsumer concepts for each NE, Wikipedia category links are used. Categories can be useful because they classify in a kind of hierarchy all the Wikipedia articles. So, in a sense, they can be considered as subsumers of the NEs to which they are linked.

However, Wikipedia categories suffer from several problems that limit their usefulness. For example "The Sagrada Familia" article is categorized as *Antoni Gaudí buildings*, *Buildings and structures under construction*, *Churches*, *Visitor attractions in Barcelona*, *Basilica churches*, etc. These categories are too complex to be directly used as subsumer concepts (i.e., it is not probable that any of them matches directly with ontological classes) and some previous analysis is needed. So, the key concepts of each category have to be detected. To extract the main concepts of each category (e.g., Basilica, churches, etc.) the same linguistic parsing detailed in section III.A is applied in order to extract Noun Phrases and core concepts which are finally matched to ontological classes.

Another limitation of Wikipedia categories is the fact that they do not always contain enough concepts to perform the matching among them and ontological classes. In these cases it can be necessary to recursively navigate to higher level categories. Wikipedia structure of categories, however, does not fulfil the strict subsumption definition, as it is more a folksonomy than a taxonomy. In order to avoid an excessive taxonomic decontextualization resulting from the recursive navigation through Wikipedia categories, only two levels have been considered.

C. Computational cost

The computational cost of the proposed method depends on the number of queries performed because they are the most expensive task [7]. We can distinguish five different tasks in which queries are performed: NE detection, NE filtering, subsumer concepts extraction, semantic disambiguation and class selection.

Both plain text and semi-structured text analyses have the same cost for NE filtering, semantic disambiguation and class selection. On one hand, to rank NEs for the relevance filtering step, two queries are needed for each NE (i.e., *2n*, where *n* represents the number of NEs). On the other hand, class selection requires as many queries as candidates a NE has (i.e. *n(c/n)2*, where *c* is the total number of candidates). For semantic disambiguation only one query is needed for each candidate (i.e. *c*).

So, the difference in computational cost between plain text analyses and semi-structured ones is in NE detection and subsumer concepts extraction. In the first approach six queries are performed to discover subsumer concepts by means of Hearst Patterns (*6n*). In the second approach, no queries are needed because NEs are directly extracted from the tagged text.

Thereby, the number of queries needed to analyse plain text is *8n+3c*, whereas only *2n+3c* are needed when dealing with Wikipedia articles. This shows how the exploitation of Wikipedia's structure aids to improve the performance of the method.

IV. EVALUATION

In this section some evaluation results are presented. We picked up as case studies the Barcelona[1] and Canterbury[2] Wikipedia articles which describe these cities. Final feature annotations are performed taking into account the space.owl[3] ontology, which maps concepts about tourist structures, geographic and geopolitical entities, types of places, etc. The evaluation is performed by analysing the articles both as plain text and also taking profit of Wikipedia semi-structure. So, in both cases the analysed content is the same.

The precision and Recall of each test have been computed. In order to calculate them, a domain expert has selected which of the features included in the articles are relevant for the city and which concepts in the ontology are the more adequate to annotate them. The *Recall* is calculated by dividing the number of correct annotations by the total of annotations the system should have annotated. The *Precision* is the number of correct annotations divided by the total number of annotations.

Fig 2 shows a comparison between the two methods (plain text and semi-structured) when applied to the cities of Barcelona and Canterbury. In the left column, the influence of threshold for filtering NE (T1) and the threshold for selection annotation (T2) are studied for Barcelona, while the right column depicts the analysis for Canterbury.

The results show that the method is able to extract, in both cases, more than a 50% of the features marked for the domain expert. We also observe that the precision tend to keep or to improve when taking into account the semi-structure of the Wikipedia articles while the recall decreases. This is because in the first approach the whole textual content is analysed. This implies that there are more possibilities to detect representative features whereas the precision may be lower because there is a higher amount of unrepresentative features which add noise to the final results. On the opposite, using the second approach the set of analysed entities is limited to those manually annotated but, in contrast, the precision is higher because the potential candidates for each feature are extracted from Wikipedia categories (tagged and selected manually by a big community of users). It is important to note that, in any case, the analysis of Wikipedia articles is, as discussed in section III.C, considerably faster than text, as the degree of analysis required to extract and annotate entities is reduced.

Considering that the final goal of the method is to enable the application of data analysis methods (such as clustering) a high precision would be desirable, even at the cost of a reduced recall. In these cases, selection thresholds can be tuned for a high precision establishing a more restrictive value.

V. RELATED WORK

In the context of the development of the Semantic Web, several methods have been proposed to detect and annotate relevant entities from electronic resources. First, manual approaches provided tools to assist the user in the annotation process (such as Annotea [16]). On the contrary, some authors tried to automate some of the stages of the annotation process to overcome the bottleneck of manual solutions. Melita[17] is based on user-defined rules and pre-defined annotations, which are employed to propose new annotations. Supervised approaches, however, are difficult to apply, due to the effort required to compile a large and representative training set. Other systems like KnowItAll [18] rely on the redundancy of the Web to perform a bootstrapped information extraction process. The confirmation of the correctness of the obtained information is requested to the user to re-execute the process.

Completely automatic and unsupervised annotation systems are scarcer. SemTag [19] performs automated semantic tagging based on the Seeker platform for text analysis. It has been able to tag a large number of pages with the terms included in a domain ontology named TAP. This ontology contains lexical and taxonomic information about areas like music, movies, sports and health. SemTag detects the occurrence of entities related to these issues in Web pages. It disambiguates the retrieved terms by using neighbour tokens and corpus statistics, picking the best label for a token. From the applicability point-of-view, Pankow [13] is the most promising system. It uses a range of well-studied syntactic patterns to mark-up candidate phrases in Web pages. Its context driven version, C-Pankow [7], improves its computational efficiency by reducing the number of queries to the search engine. However, the final association between text entities and the classes of an input domain ontology is not addressed.

VI. CONCLUSIONS

The methods proposed in this paper aim to ease the application of data-mining algorithms focused on the analysis of textual features. Applying several lexico-syntactic and statistical techniques, the proposed methods are able to extract and annotate relevant features from textual sources, exploiting, if available, semi-structured information such as user-based word tagging. In this last case, the accuracy of the results and the method's performance can be potentially improved.

As future work, we plan to adapt and test the method with other types of semi-structured resources, extending the evaluation to other domains and ontologies.

ACKNOWLEDGMENTS

This work has been partially supported by the Universitat Rovira i Virgili (predoctoral grant of C.Vicient, 2010BRDI-06-06), the Spanish Ministry of Science and Innovation (DAMASK project, Data mining algorithms with semantic knowledge, TIN2009-11005) and the Spanish Government (PlanE, Spanish Economy and Employment Stimulation Plan).

[1] http://en.wikipedia.org/wiki/Barcelona
[2] http://en.wikipedia.org/wiki/Canterbury
[3] http://deim.urv.cat/~itaka/space.owl

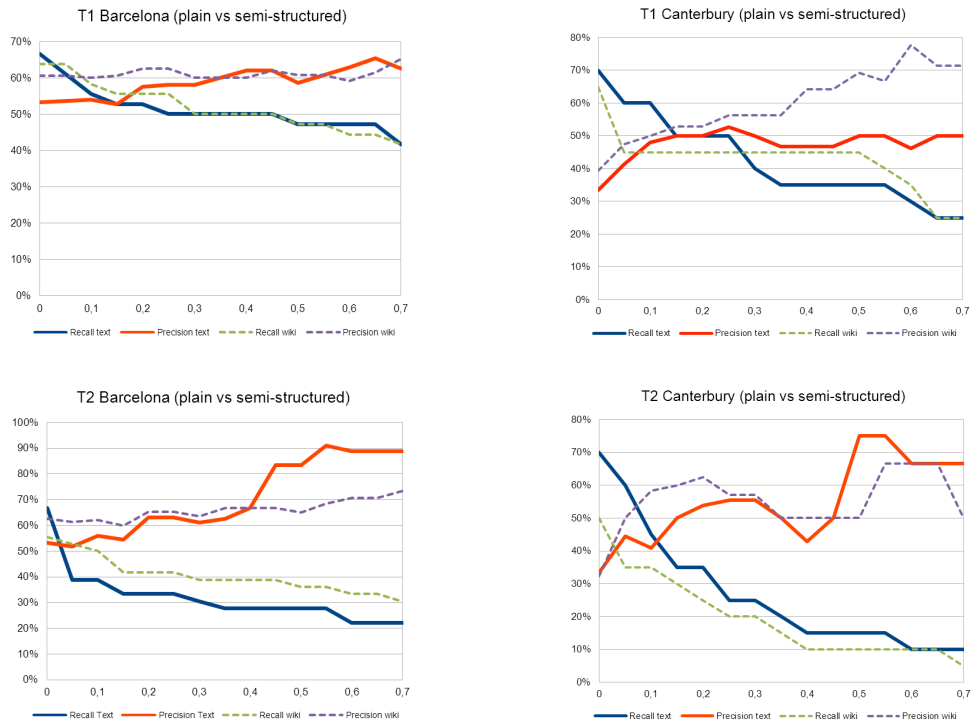


Figure 2. Evaluation Barcelona and Canterbury textual articles according to threshold values and type of analysi

REFERENCES

[1] M. Batet, A. Valls, K. Gibert, and D. Sánchez, "Semantic clustering using multiple ontologies," in 13th International Conference on the Catalan Association for Artificial Intelligence, 2010, pp. 207-216.

[2] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web - A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities," Scientific American, vol. 284, pp. 34-43, 2001.

[3] R. L. Cilibrasi and P. M. B. Vitányi, "The Google Similarity Distance," IEEE Transactions on Knowledge and Data Engineering, vol. 19, pp. 370-383, 2006.

[4] K. Church, W. Gale, P. Hanks, and D. Hindle, "Using Statistics in Lexical Analysis," in Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon, 1991, pp. 115-164.

[5] C. Leacock and M. Chodorow, "Combining local context and WordNet similarity for word sense identification," in WordNet: An electronic lexical database, 1998, pp. 265-283.

[6] P. D. Turney, "Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL," in 12th European Conference on Machine Learning, ECML 2001, 2001, pp. 491-502.

[7] P. Cimiano, G. Ladwig, and S. Staab, "Gimme' The Context: Context-driven Automatic Semantic Annotation with C-PANKOW," in 14th international conference on World Wide Web, 2005, pp. 462 - 471.

[8] D. Sánchez, D. Isern, and M. Millán, "Content Annotation for the Semantic Web: an Automatic Web-based Approach," Knowl. Inf. Syst.,, vol. 27, pp. 393-418, 2010.

[9] M. Fleischman and E. Hovy, "Fine grained classification of named entities," Proceedings of the 19th international conference on Computational linguistics - Volume 1, 2002.

[10] A. Mikheev and S. Finch, "A workbench for finding structure in texts," Proceedings of the fifth conference on Applied natural language processing, 1997.

[11] M. Pasca, "Acquisition of categorized named entities for web search," Proceedings of the thirteenth ACM international conference on Information and knowledge management, 2004.

[12] S. Lamparter, M. Ehrig, and C. Tempich, "Knowledge Extraction from Classification Schemas," in the Int. Conf. on Ontologies, Databases and Applications of SEmantics (ODBASE), 2004, pp. 618-636.

[13] P. Cimiano, S. Handschuh, and S. Staab, "Towards the self-annotating web," in 13th international conference on World Wide Web, WWW 2004, 2004, pp. 462 - 471.

[14] M. A. Hearst, "Automatic acquisition of hyponyms from large text corpora," in 14th conference on Computational linguistics - Volume 2, COLING 92, 1992, pp. 539 - 545.

[15] B. Rozenfeld and R. Feldman, "Self-supervised relation extraction from the Web," Knowl. Inf. Syst., pp. 17-33, 2008.

[16] M.-R. Koivunen, "Annotea and Semantic Web Supported Collaboration (invited talk)," in Workshop on End User Aspects of the Semantic Web at 2nd Annual European Semantic Web Conference, UserSWeb 05 2005, pp. 5-17.

[17] F. Ciravegna, A. Dingli, D. Petrelli, and Y. Wilks, "User-system cooperation in document annotation based on information extraction," in 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web, 2002, pp. 122-137.

[18] O. Etzioni, M. Cafarella, D. Downey, A. Popescu, T. Shaked, S. Soderland, et al., "Unsupervised named-entity extraction form the Web: An experimental study," Artificial Intelligence, vol. 165, pp. 91-134, 2005.

[19] S. Dill, N. Eiron, D. Gibson, D. Gruhl, R. Guha, A. Jhingran, et al., "A case for automated large-scale semantic annotation," Web Semantics: Science, Services and Agents on the World Wide Web, vol. 1, pp. 115-132, 2003