Carlos Vicient Monllaó

MOVING TOWARDS THE SEMANTIC WEB: Enabling new technologies through the semantic annotation of social contents

Ph. D. Thesis

Supervised by
Dr. Antonio Moreno

Department of
Computer Science and Mathematics

UNIVERSITAT ROVIRA I VIRGILI

December, 2014

UNIVERSITAT ROVIRA I VIRGILI

I STATE that the present study, entitled "*Moving towards the Semantic Web: enabling new technologies through the annotation of social contents*", presented by Carlos Vicient Monllaó for the award of the degree of Doctor, has been carried out under my supervision at the Department of Computer Engineering and Mathematics of this university.

Tarragona, November 21st, 2014

Doctoral Thesis Supervisor

Dr. Antonio Moreno Ribas

.

Acknowledgements

The work developed in this PhD has been partially supported by two Spanish research projects: DAMASK (DAta Mining Algorithms with Semantic Knowledge, TIN2009-11005, from January 2010 to July 2013) and SHADE (Semantic and Hierarchical Attributes in DEcision making, TIN2012-34369, from February 2013 to February 2016). These projects were funded by the Spanish Ministry of Science and Innovation and the Spanish Government (PlanE, Spanish Economy and Employment Stimulation Plan).

The author has been supported by a research predoctoral grant of the Universitat Rovira i Virgili (2010BRDI-06-06).

I would like to express my gratitude to the director of this thesis, Dr. Antonio Moreno for its guidance and advice during the elaboration of this dissertation as well as to Dr. David Sánchez who supported me throughout the beginning of this project.

I would also want to express my warm thanks to all my colleagues (Sergio, Lucas, Luis and Montse) who have been an inspiration on how to make things and who have been supportive in every way.

A special thanks to my family and to my girlfriend for encouraging me in the stressful moments.

Abstract

The advent of the Social Web (or Web 2.0) has caused an exponential growth of the documents available through the Web, making enormous amounts of textual electronic resources available. Users may be easily overwhelmed by such amount of contents and, therefore, the automatic analysis and exploitation of all this information is of great interest to the data mining community. Data mining algorithms (e.g., classification or clustering methods) exploit features of the entities in order to characterise, group or classify them according to their resemblance. Data by itself does not carry any meaning; it needs to be interpreted to convey information. In the past, classical data analysis methods were designed to deal with numerical information and they did not aim to actually "understand" the content. Thus, the data were treated as meaningless numbers and statistics were calculated on them to build models, and the interpretation of the results was left to human domain experts. In recent years (especially motived by the new paradigm of the Semantic Web, introduced by Tim Berners-Lee (Berners-Lee & Hendler, 2001) as a new global initiative), many researchers have proposed semantic-grounded data classification and clustering methods that are able to exploit textual data at a conceptual level. However, these methods usually rely on pre-annotated inputs, in which text has been manually mapped to their formal semantics according to one or several knowledge structures (e.g. ontologies, taxonomies), to be able to semantically interpret textual data such as the content of Web pages (Hotho, Maedche, & Staab, 2002).

Since the usability of all these methods is closely related to the linkage between data and its meaning, this work focuses on the development of a general methodology able to detect the most relevant features of a particular textual resource finding out their semantics (i.e. associating them to concepts modelled in a background ontology) and detecting its main topics. The methods developed in this work are not only automatic and unsupervised (avoiding the bottleneck introduced by the manual semantic mapping process), but also domain-independent (applicable to resources related to any area of knowledge) and flexible (in the sense of being able to deal with heterogeneous resources, either analysing raw text or taking profit of the semi-structure of user-generated documents such as long and complex Wikipedia articles or short and noisy Twitter tweets). The proposed methods have been evaluated with some datasets of different fields (Tourism, Oncology and Cinema), obtaining encouraging results. Thus, this work is a first step towards the automatic semantic annotation of documents, needed to pave the way towards the Semantic Web vision.

Contents

List of figures

List of tables

Chapter 1
Introduction

Since Tim Berners Lee proposed the Web for the very first time in 1989, it has been continually evolving towards the current Social Web or Web 2.0. In the Social Web, users have become both content consumers and producers. This fact, together with the proliferation of social networks, has led to an exponential growth of the contents available on the Web. These contents, either inherited from static Web 1.0 pages or user-generated in the Web 2.0, may range from merely plain text documents to more complex Web resources which present some kind of structure. While traditional Web pages were static interlinked hypertext documents which basically contained raw text or multimedia files, new Web documents are substantively different from prior Web ones, allowing all users to freely contribute and collaborate with each other in a social media dialogue as creators and consumers of user-generated content in a virtual community. This new scenario presents some interesting points:

- Dynamic and permanently updated content enriches the user experience.
- Information flows bidirectionally between site owners and site users by means of evaluation, review, and commenting.
- Site users may add content for others to see, comment, modify and improve (crowdsourcing).
- Web 2.0 sites develop APIs to allow automated usage (e.g. by individual apps or by sites that gather data from different resources and build an aggregated mashup).
- One of the most important key features of the Web 2.0 is that users have the ability to collectively classify information by means of tags, developing free taxonomies of information called folksonomies.

User-generated tags are usually quite short (even single word descriptions) and their usage is mainly oriented to the improvement of search and information retrieval without the reliance on a fixed set of pre-established categories. Some Web 2.0 examples of this technology are Wikipedia (each entry is classified by means of user-created categories), blogs (published posts are usually tagged by the author with keywords that reflect the main topics of the content) and micro-blogs (tweets may be freely annotated with hashtags). Notice that each social network

has been designed to satisfy different needs and each one presents different advantages and drawbacks. For example, Wikipedia contents are created and maintained by a large community of users and it is very hard to standardise the choice of categories (future chapters explain some consequences of this fact). In contrast, blogs are usually created and maintained by a low number of users and the choice of tags used to classify all the posts can indeed be standardised, facilitating the access of the contents per areas or topics. On the other hand, micro-blogging services such as Twitter are used by hundreds of millions of heterogeneous users and it would be impossible to reach any kind of collective agreement on the choice of hashtags. Moreover, while Wikipedia and blogs are intended for publishing long-term contents, micro-blogs updates are very volatile and hashtags evolve very quickly in time, hampering even more a potential standardization.

The analysis and categorisation of all the information freely provided by the users in the Web 2.0 represents an exciting and productive new area of study, which may have a real economic value on practical problems. For instance, large companies are making heavy investments towards the dynamic analysis of the opinions of the customers about their products on the Web, trying to detect trends that drive sales, consumer satisfaction and corporate profits (Jansen, Zhang, Sobel, & Chowdury, 2009). Governments are also very interested in the analysis of the public opinion on different social and economic issues on the Web (Tumasjan, Sprenger, Sandner, & Welpe, 2010). At a more academic level, knowledge-based and Information Retrieval systems are being developed to exploit all the potential information available on Big Data (Boyd & Crawford, 2012) (available in many fields such as Biology, Economy, Social Networks, Astronomy, Complex Networks, Health Care, etc.) and Human-Computer Interaction and Knowledge Discovery applications try to facilitate a personalised visualization of complex contents to users in a user-friendly interface (Stojanovski, Dimitrovski, & Madjarov, 2014).

In particular, the field of Knowledge Discovery (KD) provides all the data acquisition, representation, filtering, analysis and mining techniques which constitute the starting point for these areas of study. These methods permit to analyse data from different perspectives and to summarise it into useful knowledge. Technically, data mining has been generally considered as the process of finding correlations or patterns among dozens of fields in large relational databases. Nowadays it is not necessary to turn to huge, structured databases to have large data volumes to analyse. The underlying idea is that data conveys information about the patterns, associations and relationships among all the domain elements, which, in turn, can be converted into knowledge about historical patterns and future trends. The traditional method of turning data into knowledge relies on the manual analysis and interpretation carried out by expert analysts that become intimately familiar with the data and serve as an interface between the data and the users. Unfortunately, this manual approach is slow, expensive and highly subjective, and it has become impracticable due to the dramatically huge volumes of data on the Web 2.0. For all these reasons there is an urgent need for a new generation of computational theories and tools to assist humans in the

extraction of useful knowledge from raw data. It has been widely stated that one of the most important and challenging problems in data mining is the incorporation of domain knowledge (Fayyad, Piatetsky-shapiro, & Smyth, 1996), and new semantic data mining methods do necessarily have to incorporate it. Fortunately, Tim Berners-Lee (Berners-Lee & Hendler, 2001) proposed in 2001 a new global initiative for the World Wide Web: the Semantic Web (also known as Web 3.0) which promises to offer solutions to formally capture and efficiently use the domain knowledge.

The Semantic Web is a collaborative movement led by international standard bodies like the World Wide Web Consortium (W3C), whose main purpose is driving the evolution of the current Web by enabling users (and automated agents) to find, share, and combine information more easily, encouraging the inclusion of semantic content in Web pages. Semantic annotations, which specify the meaning of Web page elements (text, images, tables, videos), should be machine readable so that computers can interpret them and perform the complex work involved in finding, combining, and acting upon information on the Web. Currently machines cannot easily accomplish all of these tasks without human direction since Web pages are designed to be read and processed by people, leaving users in charge of the interpretation of semantics according to the context of the text or the domain in which it is framed. In other words, users have the ability of understanding contents whereas machines do not. The main idea of the Web 3.0 is the addition of metadata to Web pages making them machine readable. These metadata link natural language terms or multimedia resources with their semantics, i.e. metadata add conceptual descriptions to the Web page elements. Linked data are empowered by technologies such as the representation languages RDF (Resource Description Framework) and OWL (Web Ontology Language), standard terminologies such as SKOS (Simple Knowledge Organization System) and FOF (Friend Of A Friend), or query languages like SPARQL (SPARQL Protocol and RDF Query Language). Considering this new paradigm, the research community is now turning towards the development of semantic data mining techniques based on Semantic Web technologies. Such techniques focus on relations between objects in addition to the analysis of the main features/attributes of the objects themselves (Kiefer, Bernstein, & Locher, 2008). However, most of the proposed semantic data mining approaches assume already the availability of semantic Web data, which cannot be expected realistically in the short term. In fact, nowadays the great majority of Web 2.0 documents lack any kind of semantic information.

In this work we identify the importance of automatic semantic annotation mechanisms, which may help to provide meaning to the current huge and heterogeneous resources generated in the Web 2.0 so that they can not only be leveraged on the forthcoming Semantic Web but may also be already used by semantic data mining procedures. The main objective of this thesis is the design of automatic, unsupervised, domain-independent and flexible methods well suited for the semantic analysis of different sorts of Web resources, including raw text documents (inherited from the Web 1.0) as well as semi-structured Web 2.0 social content, ranging from long and detailed Wikipedia articles to short and noisy tweets. These unsupervised methodologies should be able to generate an annotated

output, enabling the feasibility of use of current semantic data mining procedures that rely on pre-defined and pre-annotated inputs.

The work presented in this dissertation describes two basic approaches to semantic data management. The first one focuses on the ontology-based identification of the main features of an electronic resource. The method may be applied to purely textual documents, but it is flexible enough to be adapted towards the extraction of features of semi-structured documents, like Wikipedia pages, leveraging the extra information available on this kind of resources. This approach starts with the detection of the Named Entities that describe a certain object of study, applying Natural Language Processing techniques. A second filtering step, in which Web-based statistics are employed, determines which of them are relevant enough. A final semantic annotation step, that employs techniques like linguistic patterns, links the remaining Named Entities with the appropriate concepts in a domain ontology, providing them with a specific meaning. This methodology has been tested with the analysis of different kinds of Wikipedia pages (tourist destinations), considering them both as simple text and as structured content. In the latter case, extra information like Wikipedia links and Wikipedia categories are used by the system to improve its performance. This new methodology of ontology-based information extraction is mainly reported in Chapter 4, in which the use of the information obtained by the method to develop a personalised recommender of touristic activities is also shown.

The second approach aims to deal with semi-structured social resources that present a reduced context and in which the previous methodologies cannot be applied. We have focused our attention on micro-blogging services, specifically on Twitter. From the reduced amount of information provided in a tweet, we have studied the use of hashtags to categorise the relevant topics of a set of tweets, so that the tweets related to a given topic could be semantically annotated. We have defined a mechanism that analyses the hashtags used in a tweet data set, links them to concepts in WordNet (in a semantic annotation step supported by Wikipedia categories) and then applies ontology-based semantic similarity measures to cluster the hashtags in topically-related groups. A case study on medical tweets related to Oncology is reported on Chapter 6.

The work developed in this PhD thesis was framed within the scope of two Spanish research projects: DAMASK (DAta Mining Algorithms with Semantic Knowledge, TIN2009-11005, from January 2010 to July 2013) and SHADE (Semantic and Hierarchical Attributes in DEcision making, TIN2012-34369, from February 2013 to February 2016). These projects were funded by the Spanish Ministry of Science and Innovation and the Spanish Government (PlanE, Spanish Economy and Employment Stimulation Plan). DAMASK proposed 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 current SHADE project intends to define new decision making methods based on outranking relations that may be used when the attributes that define the objects

have an underlying hierarchical structure. The research work shown in this dissertation was also partially supported by a research grant of the Universitat Rovira i Virgili (2010BRDI-06-06).

1.1 Objectives

The specific objectives of the thesis can be summarised as follows:

1. To make a state of the art on the current methods on ontology-based information extraction.
2. To design and implement a novel automatic, unsupervised and domain-independent method that is able to extract the most relevant features from a variety of documents, ranging from completely plain textual data to semi-structured information like Wikipedia articles.
3. To associate the identified features with ontological concepts, providing a full method for automatic semantic annotation of Web resources.
4. To study the current works on analysis of information on micro-blog social networks (more specifically, on Twitter), to identify the main characteristics of short noisy updates and to clarify the challenges associated to the automatic semantic study of this kind of user-generated contents. In particular, the factors that make inapplicable the classical methods of textual analysis (natural language processing techniques, stemming, linguistic patterns, Web-scale statistics) should be identified and new techniques able to manage them should be defined.
5. To design and developed a new automatic, unsupervised and domain-independent methodology capable of analysing a collection of tweets and categorising them in different topics, by making a semantic clustering of the hashtags they contain and filtering, from a large and noisy initial hierarchical clustering, those classes that are really relevant.

1.2 Contributions

The two main specific contributions of this Ph.D. thesis towards the fulfillment of these objectives are the following:

1. **A new unsupervised, automatic and domain-independent framework for extracting the most relevant features from a range of textual**

documents and semantically annotating them with the support of a domain ontology.

We present a methodology whose aim is to discover those features modelled in an input ontology that can be found in a textual document describing an entity. This mechanism can be adapted to exploit different kinds of resources such as plain text documents and semi-structured resources (Wikipedia articles). It has been published in the following journal:

Vicient, C., Sánchez, D., Moreno, A. (2012). An automatic approach for ontology-based feature extraction from heterogeneous textual resources. Engineering Applications of Artificial Intelligence.

The information extraction mechanisms developed in this thesis were applied on the data of 150 tourist destinations in the context of the DAMASK research project, in which a personalised semantic recommender was developed. This work was reported in the following journal:

Moreno, A., Valls, A., Martínez, S., Vicient, C., Marín, L., & Mata, F. Personalised recommendations based on novel semantic similarity and clustering procedures. AI Communications. Accepted for publication, in press.

Preliminar work on this topic was presented in the following international conferences:

Vicient, C., Sánchez, D., Moreno, A. (2011). Ontology-Based Feature Extraction. In Workshop on 4th Natural Language Processing and Ontology Engineering (NLPOE 2011) in IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT 2011) (Vol. 3, pp. 189–192). Lyon (France).

Vicient, C., Sánchez, D., Moreno, A. (2011). A Methodology to Discover Semantic Features from Textual Resources. In Sixth International Workshop on Semantic Media Adaptation and Personalization (SMAP 2011) (pp. 39–44). Vigo (Spain)

Moreno, A., Valls, A., Mata, F., Martínez, S., Marín, L.,Vicient, C. (2013). A semantic similarity measure for objects described with multi-valued categorical attributes. Series Frontiers in Artificial Intelligence and Applications (Vol. 256, pp. 263–272).

2. **A new automatic, unsupervised and domain-independent method for extracting relevant topics from micro-blogging messages.**

   We present a new methodology which focuses its attention in the automatic discovery of the topics associated to a set of tweets, based on the semantic clustering of hashtags. It is being considered for publication in the following journal:

Vicient, C., Moreno, A. (2014). Unsupervised topic discovery in micro-blogging networks. Expert Systems with Applications (under review).

Preliminar work on this topic was presented in the following international conferences:

Vicient, C., Moreno, A. (2013). A Study on the Influence of Semantics on the Analysis of Micro-blog Tags in the Medical Domain. (A. Cuzzocrea, C. Kittl, D. E. Simos, E. Weippl, & L. Xu, Eds.) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 8127, pp. 446–459.

Vicient, C., Moreno, A. (2014) Unsupervised semantic clustering of Twitter hashtags. Proceedings of the 21st European Conference on Artificial Intelligence (ECAI-2014), Eds. T.Schaub et al., pp. 1119-1120. Prague, Czech Republic, August 2014.

1.3 Document structure

The present document is divided into the following chapters:

- Chapter 2 presents the basic techniques, knowledge structures and tools used in this work. It contains brief introductions to general areas like natural language processing and automated clustering, as well as succinct descriptions of knowledge structures like Wikipedia or WordNet.
- Chapter 3 details the basic Information Extraction notions and describes related works in this area, focusing on Ontology-Based Information Extraction methods which use domain ontologies to drive the extraction process and to model the retrieved data.
- Chapter 4 proposes and evaluates a new automatic, unsupervised and domain-independent feature extraction methodology, which aims not only to discover the main characteristics that describe a certain object but also to annotate them with respect to a domain ontology. It is explained how to analyse different types of Web resources, taking profit of the additional information provided by semi-structured ones like Wikipedia articles.
- Chapter 5 presents the previous works related to the analysis of information from microblogs, focusing on the issue of topic detection and discussing the main advantages of the incorporation of semantic information that may help to alleviate the lack of context of this kind in Social Web resources and to categorise better the information they provide.

- Chapter 6 details and evaluates a new automatic, unsupervised and domain-independent method that is able to annotate semantically the hashtags available in a set of tweets, using Wikipedia as a supporting knowledge structure, in order to perform a semantic clustering procedure to detect the main underlying topics.
- Chapter 7 presents the final conclusions and comments some potential lines of future research.

Chapter 2
Background

This chapter presents the basic concepts and tools used to support the semantic methodologies developed in this work. The first part of the chapter introduces the Web as the main electronic source of information, and it comments the two basic Social Web endeavours that have been considered in this work: the Wikipedia encyclopaedia and the Twitter social network. After that, we turn our attention to the knowledge structures needed to store semantic domain information (ontologies) and, in particular, WordNet is presented as the most general-purpose English lexicon. In the last part of the chapter some key techniques are introduced: natural language analysis (including parsing, part-of-speech analysis, stemming), linguistic patterns, Web-scale statistics, ontology-based semantic similarity measures and clustering.

2.1 Work environment

This section, which presents the basic work environment (the Social Web), is divided in four parts: the conceptualisation of the Web as a corpus of data, the use of Web snippets to improve the performance of information retrieval process, and the introduction of Wikipedia and Twitter as Social Web constructions used during this dissertation.

2.1.1 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 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 characterises 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, Guthrie, & Wilks, 2003; Etzioni et al., 2004; Fayyad et al., 1996; Rosso, Montes Y Gómez, Buscaldi, Pancardo-Rodríguez, & Pineda, 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. On our work, this fact will be taken into account when we have to filter which are the more relevant Named Entities associated to the description of a given object. The redundancy of the Web, in which the same fact may be expressed in many different ways, also permits to find sentences that follow precise linguistic patterns, which are used in this work to find potential hypernyms of a Named Entity (concepts with which it may be semantically annotated).

2.1.2 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 is presented (see Figure 1). These previews, typically called snippets, despite the fact that they offer a narrow context, are informative enough to extract related knowledge without accessing the Web's content.

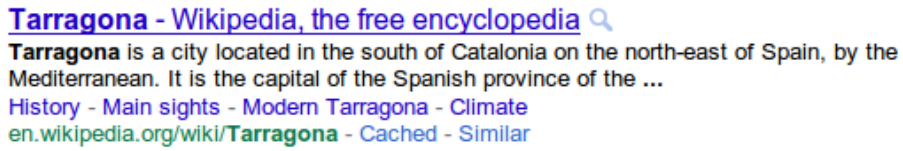


Figure 1. Snippet of a website obtained by Google for the query "Tarragona"

In this work, as will be explained in Chapter 4, snippets have been particularly useful in the pattern-based extraction of semantic annotation candidates for a given Named Entity (only considering a short context for the constructed query) and in the semantic disambiguation of terms to extract synonyms using WordNet.

2.1.3 Wikipedia

Wikipedia is a free, Web-based, collaborative, multi-lingual encyclopaedia project supported by the non-profit Wikimedia Foundation. Its 18 million articles (over 4.6 million in English, as of September 2014) have been written collaboratively by volunteers around the world, and almost all of its articles can be freely edited by anyone with access to the site.

Wikipedia is a semi-structured Web resource that includes 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 like Information Extraction, Information Retrieval and the Semantic Web, metadata are labels which give semantics to the contents that are being annotated.

Moreover, the Wikipedia is particularly useful because of its link structure. Wikipedia links bring information about relations between articles and 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 the 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 implicit information (i.e. two articles are somehow related) and that users can navigate among all related articles in an easy way.

On the other hand, category links are used to organise the knowledge contained in Wikipedia by grouping together pages on similar subjects. Categories are meant to be used as a navigational system that helps readers to quickly move from one article to a related one within a 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 category system approximates a directed acyclic graph.

For example, the Wiki about "Barcelona" has an internal link to "The Sagrada Familia" article which is categorised 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 (Figure 2). The conclusion is that Barcelona is related with Sagrada Familia, which can be categorised as a church or basilica (similar concepts), as a building (concept which is in a higher level of abstraction 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).

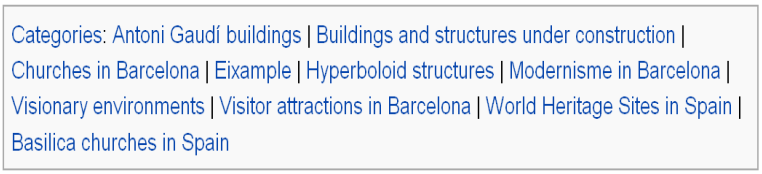


Figure 2. Example of the categories of "The Sagrada Familia"

Concretely, the use of Wikipedia in this work is twofold: to exemplify the extraction of the main features from semi-structured resources (using Wikipedia

links and Wikipedia categories, as explained in Chapter 4) and to assist the semantic annotation of Twitter hashtags that do not represent any well-defined specific concept by itself (e.g. Named Entities, acronyms, etc.), as described in Chapter 6.

2.1.4 Twitter

Twitter is one of the most famous micro-blogging systems around the world. It has more than 271 millions of monthly active users and up to 500 million Tweets are sent per day.(Twitter, 2014). Each tweet is a string of up to 140 characters (Figure 3-a) that may basically contain text, links (Figure 3-c), user mentions (Figure 3-d) and hashtags (strings preceded by the # symbol with which users tag their messages; Figure 3-b1&b2). Twitter is a real-time environment, which means that tweets contain the most up-to-date and inclusive stream of information and commentary on current events, people's opinions, business trends, etc. By contrast, they are usually ungrammatical, full of abbreviations, mood expressions and acronyms, motivating the need for systems that can extract, aggregate and categorise all its contents. Individual tweets are also very terse, often lacking sufficient context to categorise them easily into topics of interest. Moreover, a high percentage of the huge number of daily messages contains irrelevant and redundant information, which quickly lead users to a situation of information overload.

Figure 3. Parts of a Tweet

Twitterers (i.e., users of *Twitter*) may publish *tweets* to share news and opinions with the rest of the community. An important feature of Twitter that makes it different from other social networks is the fact that users do not need to give permission to the people that want to receive their messages. In fact, Twitter employs a social-networking model called "following", in which each *twitterer* can follow any other user without seeking any permission and, in consequence, he may also be followed by others without granting permission first. This is useful for users who want to receive tweets from users who they are following (i.e., *followees*) and to share their tweets with those that they are followed by (i.e. *followers*). Twitter provides an API so that researchers can retrieve sets of tweets and analyse them. Concerning the communication model, *tweets*, *replys* and *retweets* are the core of *Twitter*. Tweets are the messages published by *twitterers*. Any *twitterer* can reply to a tweet adding extra information or giving his impression about it creating a natural conversation among users. Finally, if a *twitterer* wants to only share with his *followers* a tweet that he has read he may *retweet* it and, automatically, it will be spread among his *followers*.

Last but not least, *hashtags* provide Twitter with a mechanism to semi-structure its content, as users can employ them to annotate their messages. Hashtags create a relation among tweets that share the same hashtag. Tweets that share the same hashtag are implicitly related to the same topic; thus, hashtags may, in principle, be used to categorise sets of tweets (leaving aside problems associated to polysemy or the use of acronyms, which will be commented in more depth later in this dissertation). Hashtags are also useful for retrieval purposes, as users can search for tweets containing a given hashtag in order to be aware of the news and opinions on a particular topic. Similar tagging mechanism have been employed by other Web 2.0 applications like Wikipedia which uses *wikilinks* and *category links* and classical blogs that employ simple *tags* to annotate the posts.

All of these factors define an appealing research area of study for knowledge discovery and data mining, in which traditional methodologies (as seen in future chapters) do not show a satisfactory performance and new analysis procedures must be defined.

In this work, Twitter is used to exemplify the extraction of knowledge from user-generated content in a social network (in particular, the discovery of the main topics underlying a set of tweets) in a very restricted and noisy setting that presents many challenges.

2.2 Knowledge repositories

This section is divided in two parts. Section 2.2.1 explains what is an ontology and presents its main characteristics. Section 2.2.2 briefly describes the structure of the knowledge repository WordNet and explains how to use it to obtain the synonyms, hypernyms and hyponyms of a word.

2.2.1 Ontology basics

In (Studer, Benjamins, & Fensel, 1998), an ontology is defined as a formal, explicit specification of a shared conceptualization. Conceptualization refers to the construction of an abstract model of some phenomenon in the world by identifying its basic associated concepts. 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 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 the shared understanding of some domain of interest (Uschold & Gruninger, 1996).

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

$$O = (C, \leq_C, R, \sigma_R, \leq_R, A, \sigma_A, T) \quad (1)$$

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.).

- $\leq_C$ 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).

- $\leq_R$ 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).

- σ_R: $R \rightarrow C^+$ 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_R(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.

- σ_A: $A \rightarrow C \times T$ represents the signature describing an attribute of a certain concept *C*, which takes values of a certain data type *T* (e.g. the

attribute number-of-leukocytes of the concept Blood_Analysis, which must be an integer value).

There exist different knowledge representation formalisms for the definition of ontologies. However, they all 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.

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 representation languages are KIF, RDF, KL-ONE, DAML+OIL and OWL (Gómez-Pérez, Fernández-López, & Corcho-García, 2004). There are some differences between them according to their supported degree of expressiveness. In particular, OWL is the most complete one, allowing one 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 could be possible to define, for example, a set of Symptoms which co-occur for a particular Disease using the AND operator.

Ontologies are used in different parts of the work presented in this dissertation. On the definition of semantic feature extraction methodologies (see Chapter 4) the use of the domain ontology is crucial, since it indicates the concepts of interest in the domain, with which we can annotate the relevant Named Entities in the description of an object. Only the important features for a particular domain will be annotated in the last step of the methodology, avoiding the important computational cost that would have to be assumed if we wished to annotate all the entities that appear in the analysed text. On the second part of the work, in which a semantic treatment of tweets is proposed (see Chapter 6), the central part is the use of ontology-based semantic similarity measures (on the WordNet ontology) to make a semantic clustering of the hashtags. On the topic detection algorithm WordNet is also used to compute the semantic centroid of each set of hashtags, in order to provide an appropriate label to each of the discovered topics

2.2.2 WordNet

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.

WordNet[1] is the most commonly used online lexical and semantic repository for the English language. Many authors have contributed to it (Daude, Padro, & Rigau, 2003) or used it to perform many knowledge acquisition tasks. Concretely, it offers a lexicon, a thesaurus and semantic linkage between the majority of English terms. It seeks to classify words into categories and to inter-relate 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 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.

[1] http://wordnet.princeton.edu Last access: November 10th, 2014

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 summarises the meaning of a specific concept and models the knowledge available for a particular domain. Table 1 depicts the WordNet 2.1 database statistics (number of words, synsets and senses).

Table 1 WordNet 2.1 database statistics

POS	**Unique Strings**	**Synsets**	**Total Word-Sense Pairs**
Noun	117.097	81.426	145.104
Verb	11.488	13.650	24.890
Adjective	22.141	18.877	31.302
Adverb	4.601	3.644	5.720
Totals	*155.327*	*117.597*	*207.016*

In this work, WordNet will be particularly useful to link terms with its meaning (semantic annotation) in order to be able, for example, to extract similar terms for a given term exploiting its hyponyms, hypernyms, and synsets. This will be beneficial in order to increase the set of possible annotation candidates for a given Named Entity improving the matching process in the feature extraction algorithm (see section 4.1.4.2.2). For example, Figure 4 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. WordNet is also a basic component of the algorithm for topic detection in micro-blogs, as hashtags are mapped into WordNet synsets and an ontology-based similarity measure is applied on them to build clusters of conceptually similar hashtags. Moreover, WordNet is also used to compute the semantic centroid of each of the discovered topics.

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 4. Information extracted from WordNet when querying church

2.3 Techniques

This section describes the basic ideas underlying the main Artificial Intelligence techniques used in this work, to provide the background necessary to understand the rest of the work. First, the main techniques in Natural Language Processing are presented in section 2.3.1. In section 2.3.2, Hearst linguistic patterns and their applicability to detect hyponym/hypernym relationships are discussed. After that, the use of Web-based statistical measures and ontology-based semantic similarity measures to compute the relatedness of two terms are studied (sections 2.3.3 and 2.3.4). Finally, clustering techniques are briefly introduced in section 2.3.5

2.3.1 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 (NLP) includes the study of mathematical and computational models 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.

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 of detecting sentences, tokens and parts of speech (Text processing) and performing the syntactic analysis or Part-Of-Speech tagging. The first component is able to chunk a text in order to find its minimal parts. Once the text is chunked, the minimal pieces obtained are tagged with a Part-Of-Speech (POS) tagger. The syntactic analyser or Part-Of-Speech tagger 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 Noun Phrases (referring to Named Entities) from text, in the first step of the ontology-based feature extraction process explained on Chapter 4.
- Stemming algorithm: it 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 (Van Rijsbergen, Robertson, & Porter, 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.

The following subsections explain in more detail these points.

2.3.1.1 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, the word “don't” should be separated into the words “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 it provides syntactic information of every word in the sentence structure, but it can 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 or verb groups, but without specifying their internal structure or their role in the main sentence.

A simple example of sentence analysis is provided in Figure 5, with the sentence “The red book is a black novel”. 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

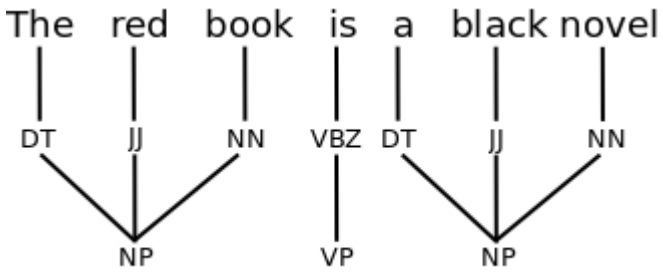


Figure 5. Sentence analysis

In this work, OpenNLP[2] 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). The parser has been used in the ontology-based feature extraction algorithm (Chapter 4) to detect the NPs present in the textual description of an object.

2.3.1.2 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.

Several algorithms, such as the lemmatisation algorithm, the stochastic algorithm, the N-gram algorithm or Porter's 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 it has been 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.

Table 2 Results of Porter stemming algorithm

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

[2] http://incubator.apache.org/opennlp/ Last access: November 10th, 2014

Table 2 shows the results of stemming the set of words of the leftmost column. 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's algorithm is based on English grammatical rules and the English word exceptions are not taken into account.

2.3.1.3 Stop words

Table 3 shows a list of stop words which includes prepositions, auxiliary verbs, adverbs, etc.

Table 3 Stop words list

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”

2.3.2 Linguistic patterns

Instance-Concept relations are called “is-a” relationships. There exist many approaches for detecting this kind of relations. This work will consider unsupervised, domain-independent ones. As stated in (Cimiano, Handschuh, & Staab, 2004), three different learning paradigms can be exploited. First, some approaches rely on the document-based notion of term subsumption (Sanderson & Croft, 1999). Secondly, some researchers claim that words or terms are semantically similar to the extent to which they share similar syntactic contexts (Bisson, Nédellec, & Cañamero, 2000; Caraballo, 1999). 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 (Ahmad, Tariq, Vrusias, & Handy, 2003; Berland & Charniak, 1999).

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 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 the effectiveness of linguistic patterns, such as the ones shown in Table 4, to retrieve hyponym/hypernym relationships.

Table 4 Hearst patterns

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

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 et al., 2004) As it stated in section 2.1.1, in this work this data sparseness problem is tackled by exploiting the Web as a corpus (Paul Buitelaar, Olejnik, & Sintek, 2004; Velardi, Navigli, Cucchiarelli, & Neri, 2006).

As will be shown in Chapter 4, the feature extraction algorithm uses linguistic patterns to find the concepts potentially associated to the Named Entities describing a certain object.

2.3.3 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 (Hanson, Cowan, & Giles, 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 (J. H. Lee, Kim, & Lee, 1993; Richardson, Smeaton, & Murphy, 1994) or with a combination of statistics and lexical information, in hybrid approaches (Jiang & Conrath, 1997; Resnik, 2011). 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 (Keller, Lapata, & Ourioupina, 2002; Turney, 2001) 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 (Cilibrasi & Vitányi, 2006; Cimiano et al., 2004; Etzioni et al., 2005) 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. 2) (Church, Gale, Hanks, & Kindle, 1991) or the Symmetric Conditional Probability (SCP) (Dias, Santos, & Cleuziou, 2006).

$$PMI(a,b) = \log_2 \frac{\rho(ab)}{\rho(a)\rho(b)} \tag{2}$$

PMI statistically assesses the relation between two words (a, b) as the conditional probability of a and b co-occurring within the text. This score is derived from probability theory. Here, ρ (ab) is the probability that both terms co-occur. If they are statistically independent, then the probability that they co-occur is given by the product $\rho(a)\,\rho(b)$. If they are not independent, and they have a tendency to co-occur, then $\rho(ab)$ will be greater than $\rho(a)\,\rho(b)$. Therefore the ratio between these numbers measures the degree of statistical dependence between the terms. 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. 3 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.

$$PMI_{IR}(a,b) = \log_2 \frac{\frac{hits(a\ AND\ b)}{\#total\_webs}}{\frac{hits(a)}{\#total\_webs}\ \frac{hits(b)}{\#total\_webs}} \quad (3)$$

In this work, in order to provide a scalable solution, a variant of this measure will be used to compute the relatedness between an object and the Named Entities appearing on its textual description (section 4.1.3) and to select the best annotation for each Named Entity, taking into account all of its potential candidates (section 4.1.4).

2.3.4 Ontology-based semantic similarity

All data mining methods rely on mechanisms that enable the comparison between two elements/attributes in order to detect their degree of alikeness. Traditional data mining methodologies have been broadly studied and there is a wide range of mathematical formulas used to compare numerical attributes. Some examples are the Euclidean distance, the cosine distance, the Manhattan distance, etc. (Deza & Deza, 2009). The analysis and classification of textual resources implies the comparison among words. Words are labels referring to concepts, which define their semantics. In consequence, semantic similarity measures are needed. Semantic similarity is precisely the science that aims to estimate the alikeness between words or concepts by discovering, evaluating and exploiting their semantics. As semantics is an inherently human feature, methods to automatically calculate semantic similarity rely on evidences retrieved from one or several manually constructed knowledge sources (for example, from ontologies). The goal is to mimic human judgments of similarity by exploiting implicit or explicit semantic evidences.

In the literature, we can distinguish different approaches to compute semantic similarity according to the techniques employed and the knowledge exploited to perform the assessment. The most prominent ones are edge counting-based measures (that map terms to ontological concepts and calculate their similarity by looking at the length of the is-a path connecting them), feature-based measures (which assess the similarity between concepts as a function of their properties, i.e. taking into account synonyms, definitions, etc.) and information-content based measures (that quantify the degree of information provided by the concepts by means of the statistical information derived from a corpus).

Evaluating and comparing different semantic similarity measures is a difficult task since the notion of similarity is subjective (Bollegala, Matsuo, & Ishizuka, 2007). In order to enable fair comparisons, several authors created evaluation benchmarks consisting on word pairs whose similarity was assessed by a set of humans. Rubenstein and Goodenough (Rubenstein & Goodenough, 1965) and Miller and Charles (Miller & Charles, 1991) define some experiments in which native English speakers assessed the similarity of 65 word pairs selected from ordinary English nouns on a scale from 0 (semantically unrelated) to 4 (highly

synonymous). The results of such experiments have become *de facto* standard benchmarks to evaluate and compare the accuracy of similarity measures. As a result, correlation values obtained against those benchmarks can be used to numerically quantify the closeness of two sets of ratings (i.e. the human judgments and the results of the computerised assessment). If the two rating sets are exactly the same, the correlation coefficient is 1 whereas 0 means that there is no relation.

In (Sánchez, Batet, Isern, & Valls, 2012a), the authors compared the performance of some of the most popular semantic similarity measures taking the correlation values originally reported by related works for the benchmarks of Rubenstein and Goodenough and Miller and Charles (when available). In case in which a concrete measure depended on certain parameters (such as weights or corpora selection/processing) the best correlation value reported by the authors was compiled. It is important to note that, even though some of them relied on different knowledge sources (such as tagged corpora or the Web), all ontology-based ones used WordNet (recall section 2.2.2). Table 5 summarises this comparative.

Table 5. Correlation values for each semantic measure.

Measure	Type	M&C	R&G	Evaluated in
Path (Rada)	Edge	0.59	N/A	(Petrakis et al. 2006)
Wu & Palmer	Edge	0.74	N/A	(Petrakis et al. 2006)
Leacock & Chodorow	Edge	0.74	0.77	(Patwardhan,Pedersen 2006)
Rodriguez	Feature	0.71	N/A	(Petrakis et al. 2006)
Tversky	Feature	0.73	N/A	(Petrakis et al. 2006)
Petrakis	Feature	0.74	N/A	(Petrakis et al. 2006)
Resnik	IC	0.72	0.72	(Patwardhan,Pedersen 2006)
Lin	IC	0.7	0.72	(Patwardhan,Pedersen 2006)
Jiang & Conrath	IC	0.73	0.75	(Patwardhan,Pedersen 2006)

Correlation values indicate that measure accuracies are very similar through the different families. However, the applicability and generality of each type of measure depend on the principle they exploit. The main advantage of edge-based measures is their simplicity. They only rely on the geometrical model of an input ontology whose evaluation requires a low computational cost. However, several limitations hamper their performance. One of these limitations is that they rely on the notion that all links in the taxonomy represent a uniform distance (Bollegala et al., 2007); thus, they require wide ontologies with a relatively homogenous distribution of semantic links and good domain coverage to minimise this problem (Jiang & Conrath, 1997). Consistent ontologies like WordNet work properly with these measures (Pirró & Seco, 2008). Feature-based measures exploit more semantic evidences than edge-counting approaches, evaluating both commonalities and differences of compared concepts, but they rely on features like glosses or synsets (in addition to taxonomic and non-taxonomic relationships); therefore, those measures can only be applied to ontologies in

which this information is available. Another problem is their dependence on weighting parameters that balance the contribution of each feature. Finally, information-content measures need an accurate computation of concept probabilities that requires a proper disambiguation and annotation of each noun found in the corpus. If either the taxonomy or the corpus changes, re-computations must be recursively executed for the affected concepts. So, it is necessary to perform a manual and time-consuming analysis of corpora and the resulting probabilities will depend on the size and nature of input corpora. Moreover, the background taxonomy must be as complete as possible in order to provide reliable results. All those aspects limit the scalability and applicability of those approaches.

Considering all the advantages and drawbacks of each one of the types of similarities, we have chosen the Wu and Palmer similarity measure to compare two different terms because it does not depend on a corpora, it does not have tuning parameters and it has a very low computational cost.

The next equation (Eq. 4) shows the semantic similarity between two concepts using the Wu and Palmer distance (Z. Wu & Palmer, 1994). The main difference with the other edge-counting measures is that whereas they omit the fact that equally distant concept pairs belonging to an upper level of the taxonomy should be considered as less similar than those belonging to a lower level (as they present different degrees of generality), Wu and Palmer takes into account the depth of the concepts in the hierarchy to avoid this problem.

$$sim_{W\&P}(c_1, c_2) = \frac{2 \times N_3}{N_1 + N_2 + 2 \times N_3} \quad (4)$$

In this expression N_1 and N_2 are the number of is-a links from c_1 and c_2 respectively to their Least Common Subsumer (LCS) in the reference ontology, and N_3 is the number of is-a links from the LCS to the root of the ontology. This measure ranges from 1 (for identical concepts) to 0 (when the LCS is the root of the ontology, so the concepts do not have any common ancestor).

2.3.5 Clustering

Clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense or another) to each other than to those in other groups. It is a main task of exploratory data mining, and a common technique for statistical data analysis, used in many disciplines, including machine learning, pattern recognition, image analysis, information retrieval, etc.

Figure 6 (based on (Jain, Murty, & Flynn, 1999)) depicts the main classification of clustering approaches. There is a distinction between hierarchical and partitional approaches. Hierarchical clustering groups data objects with a sequence of partitions, either from singleton clusters to a cluster including all

individuals or vice versa, whereas partitional methods directly divides the initial set of objects into some pre-specified number of clusters without the hierarchical structure. It is important to notice that all of the different approaches, regardless of their placement in the taxonomy, may be implemented according to two different algorithmic strategies: agglomerative and divise. An agglomerative approach begins with each object in a distinct (singleton) cluster, and successively merges clusters together until a stopping criterion is satisfied. A divisive method begins with all objects in a single cluster and iteratively splits a cluster into smaller clusters until a stopping criterion is met.

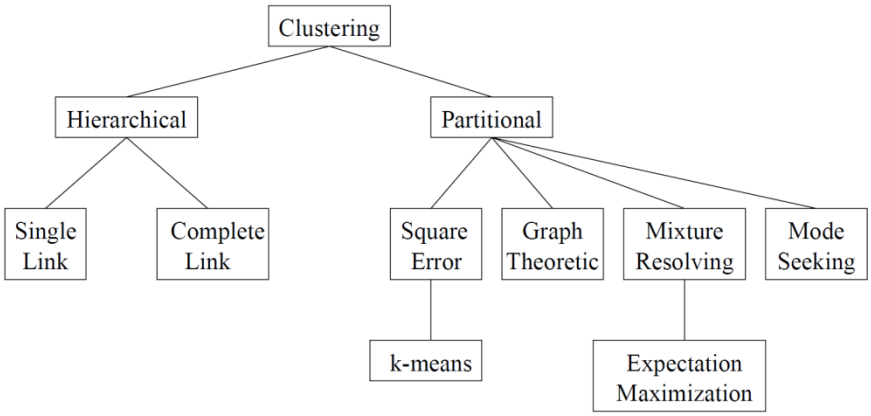


Figure 6. A taxonomy of clustering approaches

All clustering methods need a mechanism to compare objects in order to generate a similarity matrix to be the input of the methodology. However, hierarchical clustering also needs a linkage criterion to compute the distance between clusters. Most hierarchical clustering algorithms are variants of the single-link (Sneath & Sokal, 1973) and complete-link (King, 1967) algorithms. Single linkage and complete linkage consider all the points of a pair of clusters, when calculating their inter-cluster distance, and they are also called graph methods. In single-linkage the distance between two clusters is computed as the distance between the two closest elements in the two clusters (minimum distance criterion). It can produce the "*chaining*" effect, where a sequence of close observations in different groups cause early merges of those groups. Complete linkage calculates the distance between two clusters as the maximum distance between a pair of objects, one in one cluster, and one in the other (maximum distance criterion). Complete linkage has the opposite problem of single-linkage, as it might not merge close groups because of irrelevant outlier members that are far apart.

The common criticism for classical hierarchical clustering algorithms is that they lack robustness and they are, hence, sensitive to noise and outliers. Once an object is assigned to a cluster, it will not be considered again, which means that hierarchical clustering algorithms are not capable of correcting possible previous

misclassifications. The computational complexity for these clustering algorithms is at least quadratic, hampering their application to large-scale data sets.

Partitional clustering assigns a set of objects into clusters with no hierarchical structure. In principle, the optimal partition, based on some specific criterion, can be found by enumerating all possibilities; however, this brute force method is infeasible in practice, due to the prohibitive computational cost (Liu, 1968) Even for a small-scale clustering problem (organizing 30 objects into 3 groups), the number of possible partitions is $2\,x\,10^{14}$. Therefore, heuristic algorithms have been developed in order to seek approximate solutions.

Although there are several partitional approaches, the K-means algorithm is one of the best-known. It is classified as a squared error-based clustering algorithm (Forgy, 1965), (MacQueen, 1967) and it requires as input the desired number of *K* clusters, an initial assignment of data to clusters and a distance measure to compare the elements of the dataset. The K-means algorithm is very simple and can be easily implemented to solve many practical problems. By contrast, it presents many drawbacks. For example, there is no efficient and universal method for identifying the initial partitions and the optimal number of clusters *K*. It is also sensitive to outliers and noise. Even if an object is quite far away from the cluster centroid, it is still forced into a cluster and, thus, it may distort the cluster shape. The use of heuristics and the iterative procedure of K-means cannot guarantee convergence to a global optimum either.

Considering all the advantages and drawbacks of each one of the approaches, we have chosen the hierarchical clustering method (with complete linkage) to classify the hashtags contained in a set of tweets (Chapter 6). This method allows obtaining clusters of different levels of generality, and we are not forced to know (or guess) in advance which is the appropriate number of classes to obtain. The Wu and Palmer distance has been used to build the semantic similarity matrix. To deal with the problem of noise and outliers, we have proposed a mechanism able to filter the irrelevant clusters in an unsupervised way setting two different thresholds, which constrain the number of elements in a cluster and their homogeneity (this filtering method will be seen in Chapter 6). At the same time, this filtering provides a mechanism to convert the clustering hierarchy in a set of final clusters without the necessity of forcing a pre-specified number of clusters (as in partitional methods).

2.4 Summary

As seen in this chapter, the development of automatic and unsupervised analysis methods to deal with Web 2.0 resources needs a wide spectrum 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 & 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 minimise 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, a mechanism to interpret texts is also needed and the area of Natural Language Processing has been introduced. Moreover, as the feature extraction process is based on the detection of named entities (that represent real entities) and its annotation, we needed a way to find the concepts which named entities represent. Lexico-syntactic patterns, in particular Hearst Patterns, have been proposed to carry out this task.

In this work OWL ontologies are used to drive the feature extraction process indicating the concepts that we want to extract from an analysed entity in a particular domain or area of study. On the other hand, WordNet has been presented as a knowledge repository that can be used:

a) 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 do not match with any ontological class and getting synonyms, hypernyms and hyponyms the probability of ontology matching increases.

b) To perform the semantic annotation of words (link between terms and its meaning mapped as WordNet synsets) making possible a mechanism to compare different terms (e.g. hashtags) at a conceptual level.

c) To find semantic centroids (topics) from a set of annotated terms.

A review of the main semantic similarity measures, that can be used to estimate the resemblance between concepts, has been presented. Through the exploitation of knowledge offered by ontologies and the semantic measures reviewed in this chapter, it is possible to compute the similarity/distance between concepts in order to develop appropriate operators to guide the clustering process of lexical terms.

The next chapters present the first methodology developed in this work: a new unsupervised, domain-independent and flexible semantic feature extraction process. First, Chapter 3 provides a review of the state of the art on ontology-based information extraction mechanisms. After that, Chapter 4 describes this new methodology, in which repositories like Wikipedia and WordNet are exploited,

and techniques like Natural Language Processing, linguistic patterns, Web-based statistics and semantic similarity measures are used.

Chapter 3
Ontology-based Information Extraction

The first contribution of the thesis, to be described in chapter 4, is the definition of an unsupervised, domain-independent and flexible system that can analyse textual and semi-structured resources, under the guide of a domain ontology, to extract and annotate its more relevant features. This chapter discusses related works in information extraction, especially those that use the semantic knowledge stored in an ontology.

This chapter is structured as follows:

- Section 3.1 provides a brief overview of the field of Information Extraction. It also makes a comparison between traditional approaches (based on relation-dependent manually tagged examples) and open ones (that employ Machine Learning techniques to be applicable in any domain).
- Section 3.2 introduces the use of ontologies in Information Extraction, distinguishing between ontology-based and ontology-driven approaches.

3.1 Information Extraction

In the last 20 years 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, Brown, & Jablonski, 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, overlooking irrelevant words and phrases and attempting to home in on entities and the relationships between them (Etzioni, Banko, Soderland, &

Weld, 2008). These data may be directly shown to users, or stored in a database or spread sheet for direct integration with a back-office system, or they may be used for indexing purposes in search engine/Information Retrieval (IR) applications (Xiao et al., 2004). If we compare IE and IR, whereas IR simply finds texts and presents them to the user (as classic search engines do), 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, Masciari, Rende, & Tagarelli, 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. The main advantages and disadvantages of both approaches are summarised in Table 6 (Cimiano, 2006b),and they are also discussed in more detail in the following subsections.

Table 6 Comparison of traditional IE and Open IE

	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

3.1.1 Traditional IE systems

Traditional methods on IE have focused on the use of supervised learning techniques such as hidden Markov models (Freitag & Mccallum, 1999; Skounakis, Craven, & Ray, 2003), self-supervised methods (Etzioni 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 their perfomance is poor 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 was KnowItAll (Etzioni 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 information 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. KnowItAll also 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. Some recent systems that extract pre-defined types of information based on training data include StatSnowBall (Zhu, Nie, Liu, Zhang, & Wen, 2009), ExtremeExtraction (Freedman et al., 2011), NELL (Never Ending Language Learning) (Carlson et al., 2010) and PROSPERA (Nakashole, Theobald, & Weikum, 2011).

Recently there have been some authors that have proposed the use of Linked Open Data to improve the Web IE process. Concretely, in the on-going LODIE (Linked Open Data for Information Extraction) project (Ciravegna, Gentile, & Zhang, 2012; Z. Zhang, Gentile, & Augenstein, 2014) they aim to use Linked Open Data as a seed for the learning of extraction rules. Moreover, they also intend to extend their analysis not only to pure textual sources but to more structured sources, like HTML Web pages, by introducing specialised wrappers and interpreters of lists and tables.

3.1.2 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 et al., 2008). Traditional IE requires pre-specifying a set of relations of interest and then

providing training examples for each of them. *Open Information Extraction* (Open IE) (Banko & 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 utilise 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 unlexicalised 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.

One of the main precursors of this kind of systems is TextRunner (Banko & Etzioni, 2008; Etzioni 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 a sentence analyser. When the examples are fed to Machine Learning methods, domain-specific extraction patterns can be automatically learned and used to extract facts from text. Instead of demanding hand-tagged corpora, these systems require the user to specify relation-specific knowledge through a small set of seed instances known to satisfy the relation of interest, or a set of manually 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. Other similar approaches were WOE (F. Wu & Weld, 2010) and StatSnowBall (Zhu et al., 2009).

These initial systems suffered from two problems: the presence of incoherent and uninformative extractions. A second generation of open IE systems have heavily improved their perfomance by introducing new techniques. For instance, RE-VERB (Etzioni, Fader, Christensen, Soderland, & Mausam, 2011; Fader, Soderland, & Etzioni, 2011) forces a syntactic constraint and a lexical constraint on its extractions to limit the wrong results. Syntactically, the relation phrases to be studied must be either a simple verb phrase, a verb phrase followed by a

preposition, or a verb phrase followed by a noun phrase and ending in a preposition. Lexically, the relation phrase must belong to a large pre-built dictionary of relation phrases known to take many arguments. The resulting extractions are assigned a confidence score using a logistic regression classifier. An improved version of RE-VERB, called R2A2 (Etzioni et al., 2011), introduces an argument identifier, which forces arguments of the relation to belong to a short number of part-of-speech combinations, identified with a study of 250 random sentences.

The main shortcomings of RE-VERB and WOE are that they only extract relations based on verbs, and they make only a local analysis of single sentences, which may lead to the extraction of wrong facts. These two aspects were improved with a system called OLLIE (*Open Language Learning for Information Extraction*) (Mausam, Schmitz, Bart, Soderland, & Etzioni, 2012). This system uses a set of high precision seed tuples from RE-VERB to bootstrap a large training set, over which OLLIE learns *open pattern templates*, which express different ways of representing relationships (not necessarily using verbs). These patterns are used in the extraction of new facts. The system also analyses the context around the extracted tuple to add extra information (attribution, clausal modifiers) that is very valuable to assess the confidence on the correctness of the extraction.

3.2 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 a 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 the application of IE to complex domains such as functional genomics that require more text understanding pushes towards the use of more sophisticated semantic knowledge resources and thus towards ontologies viewed as conceptual models.

In recent years, *ontologies* (i.e. formal, explicit specifications of shared conceptualizations, as introduced in section 2.1.1) have emerged as a new paradigm to model and formalise domain knowledge in a machine readable way. They are designed for being used in applications that need to process the content

of information, as well as to reason about it. They permit greater machine interpretability of content than that supported by XML, RDF and RDF Schema (RDF-S), by providing an additional vocabulary along with a formal semantics. So, ontologies represent an ideal knowledge background in which to base text understanding and the extraction of relevant information. Their use 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 & 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 it. On the other hand, an annotator can commit to the ontology and annotate only the parts of the text that are relevant from the ontology's point of view.

Global scale initiatives (e.g. the Semantic Web (Berners-Lee & Hendler, 2001)) have led to the development of ontologies for many domains. Nowadays, thousands of domain ontologies are freely available through the Web (Ding et al., 2004) and big, detailed and consensued general-purpose ontologies (such as the one described in section 2.2.2, WordNet (Fellbaum, 1998)) have been developed.

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

3.2.1 Ontology exploitation for IE

IE and ontologies are related in two ways (Nédellec & Nazarenko, 2005):

- Ontologies may be used for Information Extraction: IE needs ontologies as part of the understanding process for extracting the relevant information from a document.
- Information Extraction may be used to populate, refine and improve existing ontologies: texts are useful sources of knowledge to design and enrich ontologies.

These two processes, as can be seen in Figure 7, can be combined in a cyclic fashion: ontologies are used to interpret the text at the right level for IE and IE extracts new knowledge from text, to be integrated in the ontology.

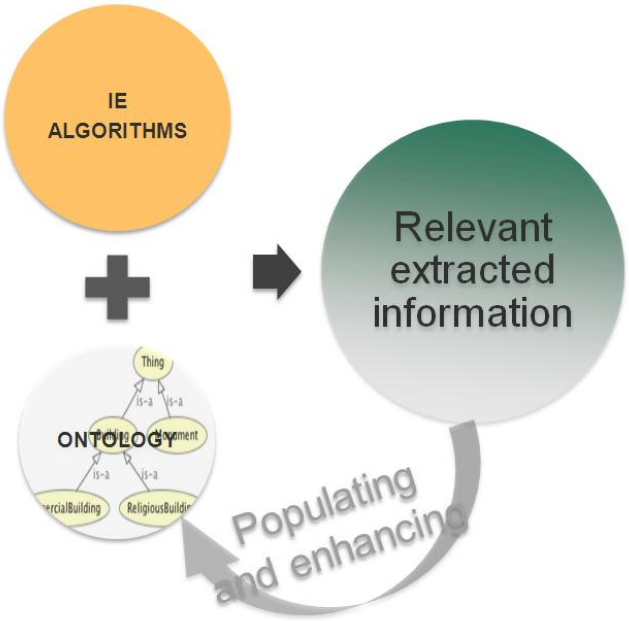


Figure 7 Ontology exploitation for IE (cyclic process)

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.

Regardless of whether one wants to use ontological knowledge to interpret natural language or to exploit written documents to create or update ontologies, 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 definition 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 the text interpretation. As such, an ontology is not a purely conceptual model, but 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 lexicalised 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 only parts of it. The ontological knowledge involved in IE can be viewed as a set of interconnected and concept-centered descriptions, or "conceptual nodes". In these conceptual nodes the concept properties and the relations between concepts are explicit. These conceptual nodes should be understood as chunks of a global model of the domain.

In general, the template or form to be filled by IE is a partial model of the world knowledge. IE forms are also classically viewed as a model of a database to be filled by the extracted instances. In (Nédellec & 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 lexico-syntactic 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. Specialised lexical or keyword lists are commonly used to identify the referential entities in documents. Three main objectives of these specialised 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 it (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 side-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 Machine Learning-based experiments have been done to exploit hierarchies of WordNet and of more specific lexicons, such as UMLS (Freitag, 1998). These 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 & Wilks, 1998), which represents an ontology as a hierarchy of concepts, each concept being associated with an attribute-value structure, or in (Embley, Campbell, Smith, & Liddle, 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, 2000).

3.2.2 Ontology-based Information Extraction

Ontology-based IE systems as those approaches relying on predefined ontologies in one or several stages of the extraction process(Wimalasuriya & Dou, 2010). 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 often 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.

It can be noticed that this 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, 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 ten 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, & Studer, 2003), NOMOS (Niekrasz & Gruenstein, 2006) or Vannotea (Schroeter & Hunter, 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 URLs on the Web[3] and at

[3] http://googleblog.blogspot.com.es/2008/07/we-knew-web-was-big.html Last access: November 10th, 2014

least 0.33 billion indexed pages[4], it is easy to envisage the unfeasibility of manual annotation of Web resources.

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, Petrelli, & Wilks, 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, & Gottlob, 2001). Another preliminary work proposing semi-automating the annotation of Web resources is the work described in (Kiyavitskaya, Zeni, Cordy, Mich, & Mylopoulos, 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 & Mooney, 2003; Roberts et al., 2007). WebKB (Cafarella, Downey, Soderland, & Etzioni, 2005) and Armadillo (Alfonseca & Manandhar, 2002) use supervised techniques to extract information from Computer Science websites. Likewise, S-CREAM (Cunningham, Maynard, Bontcheva, & Tablan, 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 (P Buitelaar, Cimiano, Frank, Hartung, & Racioppa, 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 et al., 2003) performs automated semantic tagging from large corpora based on the Seeker platform for text analysis and tags a 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, Terziev, Manov, & Ognyanoff, 2004) is another example of unsupervised domain-independent system. It scans documents looking for entities corresponding to instances in its input ontology. The authors of (Faria, Serra, & Girardi, 2014).have recently proposed an automatic and domain-independent system that extracts information from the Web to generate

[4] http://www.worldwidewebsize.com/ Last access: October 20th, 2014

ontological instances. In this work the input ontology is used to automatically generate the rules needed to extract the instances from text and to classify them in the corresponding ontology classes.

Another interesting annotation application is presented in (Michelson & 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 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, & Staab, 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.

A relatively recent survey (Karkaletsis, Fragkou, Petasis, & Iosif, 2011) suggests to characterise general-purpose ontology-based information extraction systems in four categories, depending on the level of ontological knowledge that they consider: lists of domain entities (or instances), class hierarchy, properties/relations, and the whole ontology.

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, & Staab, 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 & 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 these data, one is able to use different learning approaches in a semi-supervised fashion. As a result, the process is extended. The human expert has to validate each extension before continuing.

Feilmayr et. al.(Feilmayr, Parzer, & Pröll, 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 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 (P Buitelaar 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 annotations 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 level, rules for extraction of soccer-specific entities, such as actors in soccer, teams and tournaments are implemented. Finally, data are 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).

(Z. Li & 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. After an initial basic linguistic analysis of the 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 & 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.

There are some more recent examples of domain-dependent ontology-based IE systems. For instance, Moreno et al. (Moreno, Isern, & Fuentes, 2013) use specific ontologies to guide the analysis of scientific papers in order to retrieve instances of biological regulatory networks. Çelik and Elçi use ontologies to guide the automated analysis of résumés in order to extract the different components of a curriculum, including personal data, previous studies, work experience, etc. (Çelik & Elçi, 2013).Yang et al. (X. Yang, Gao, Han, & Sui, 2012) rely on an ontology to analyse user complaints about Chinese food to try to detect potential health hazards.

3.2.3 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 & 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.
- An update of the contents of the domain ontology has a direct and immediate effect on the documents to be considered; thus, the careful maintenance and actualisation of the ontology has an extraordinary importance (Yildiz & Miksch, 2007).

One of the most well-known examples of ontology-driven information extraction systems in OntoSyphon (McDowell & 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 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 & 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) have also been proposed.

In a preliminary work of this thesis, we designed and developed an information extraction system (Carlos Vicient, 2009) guided by the classes of an input ontology, although the set of Web pages to be analysed was fixed and no Web searches were performed. The methodology was domain-independent, although 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 analysed the Wikipedia pages related to the touristic destinations in the following way:

- All the subclasses of the class were 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 were also searched in the list of categories associated to the Wikipedia page.
- The text associated to each of the images of the page was also compared with the subclasses of the class.

The numerical attributes related to the CityClass were instantiated by analysing 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 was purely syntactical.

Van Hague et al.(Hage, 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, & Pronk, 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 et al., 2005). This work is framed in the Cyc project, the ambitious effort that has been going on for some decades to formalise all the world's

commonsense knowledge. In particular, the authors 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.

3.3 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 were interested in this last option, as we wanted 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 work preliminary of this dissertation (Carlos Vicient, 2009) along the initial steps of the DAMASK project felt into this category.

The next chapter describes the new, unsupervised and domain-independent ontology-based information extraction procedure developed in this work, which aims to uncover the main features associated to a description of a given object. Natural language processing methods, Web-scale statistics, linguistic patterns and WordNet are used in this algorithm, that may be applied to both textual resources and semi-structured ones like Wikipedia articles.

Chapter 4
A semantic unsupervised domain-independent framework for extracting relevant features from a range of heter-ogeneous resources

In this chapter we propose a method that, given an input document describing an entity (e.g. a Web resource, a Wikipedia article or a plain text document) and an ontology stating which features should be extracted (e.g. touristic points of interest), it is able to detect, extract and semantically annotate relevant textual features describing that entity. The method has been designed in a general way so that it can be applied to a range of textual documents going from raw plain text to semi-structured resources (e.g. tagged Wikipedia articles). In this last case, the method is able to exploit the pre-processed input to complement its own learning algorithms. The key point of the work is to leverage the syntactical parsing and several natural language processing techniques with the knowledge contained in the input ontology, and the use of the Web as a corpus to assist the learning process to be able to 1) identify relevant features describing a particular entity in the input document, and 2) associate, if applicable, the extracted features to concepts contained in the input ontology. In this manner, the output of the system consists on tagged features which can be directly exploited by semantically grounded data analysis methods (Batet, Valls, & Gibert, 2011).

4.1 Methodology

This section provides a thorough description and formalization of the proposed methodology, whose aim is to discover those features modelled in an input ontology that can be found in a textual document describing an entity. The general algorithm is introduced in section 4.1.1, and the following three sections detail its basic components. Sections 4.1.5 and 4.1.6 explain its adaptation to the analysis of

plain text documents and semi-structured resources (Wikipedia articles). Section 4.1.7 analyses the temporal cost of the method according to the type of input. Finally, section 4.2 discusses the results obtained in the evaluation and section 4.3 summarises the main contributions of the framework.

4.1.1 General algorithm

Algorithm 1. Ontology-based feature extraction method

```
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   /* Retrieval of 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   /* Annotation of NEs with ontological classes */
19   OC  := extract_ontological_classes(do)
20   ∀ ne_i ∈ NE  {
21         /* Retrieval of  Subsumer Ontological Classes (i.e. potential
22         annotations) for each Subsumer Concept of each NE*/
23         SC  := get_subsumer_concepts_list(ne_i )
24         /*  Application of direct matching */
25         SOC  := extract_direct_matching(OC, SC)
26         /* If direct matching fails, semantic matching is applied*/
27         if |SOC | == 0 {
28               SOC := extract_semantic_matching(OC, SC, ne_i, ae)
29         }
30         /* if a similar ontological class is found, the best
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 }
```

Algorithm 1 provides a high-level pseudo-code description of the proposed algorithm. Table 7 contains the explanation of the basic elements of the algorithm.

Several of the proposed functions are abstract and can be overwritten in order to adapt the analysis to different kinds of inputs (plain text documents or semi-structured ones). Moreover, since the feature extraction process is guided by the input ontology, by using different domain ontologies (e.g. in Medicine (Spackman, 2004), Tourism (Moreno, Valls, Isern, Marin, & Borràs, 2013) or Chemical Engineering (Morbach, Yang, & Marquardt, 2007)), the system is able to adapt the feature extraction process to the domain of interest. The inputs of the system are the document to be analysed (e.g. a text describing a city), the name of the object of interest (e.g. the name of a touristic destination) and the ontology detailing the features to be extracted (e.g. touristic activities or points of interest).

Table 7 Main definitions used in the feature extraction algorithm

Element	Definition
ae *(analysed entity)*	Name of the real entity which is being analysed (e.g. Barcelona)
wd *(web document)*	Web document which contains the information about the *ae.*
do *(domain ontology)*	Domain ontology used in order to specify which kind of concepts will be taken into account during the feature extraction process.
Pd *(parsed document)*	Web document parsed in a readable format for the system.
PNE *(Potential Named Entities)*	List of potential Named Entities (entities extracted from the text that have not been checked and filtered yet).
NE (*Named Entities*)	List of selected *PNE*s, which are those with a relevancy score higher than the NE_THRESHOLD $NE = \{x \in PNE \,/\, NE\_SCORE(x, ae) > NE\_THRESHOLD\}$ (3)
SC *(Subsumer Concepts)*	Abstractions of collections of real entities which share common characteristics. Subsumer concepts and their abstractions are taxonomically related. For example, the subsumer concept of the real entities *The Sagrada Familia* and *St. Peter's Basilica* could be *basilica.*
OC *(Ontological Classes)*	List of all the classes (i.e., concepts) modelled in the input ontology
SOC *(Subsumer Ontological Classes)*	Subset of the ontological concepts that correspond to the subsumer concepts of a NE
ac *(annotated class)*	Subsumer ontological concept which represents the final annotation of the analysed NE

In order to discover the relevant features of an entity, we focus on the extraction and selection of *Named Entities* (NEs) found in the text. A NE is a Noun Phrase which refers to a real world entity and can be considered as an instance of a conceptual abstraction (e.g. *Barcelona* is a NE and an instance of the

conceptual abstraction *city*). Due to their concrete nature, it is assumed that NEs describe, in a less ambiguous way than general words, the relevant features of a particular entity (Abril, Navarro-Arribas, & Torra, 2011).

To select which of the detected NEs are the most related to the analysed entity, a relevance-based analysis relying on Web co-occurrence statistics is performed. Afterwards, the selected NEs are matched to the ontological concepts of which they can be considered instances. In this manner the extracted features are presented in an annotated fashion, easing the posterior application of semantically-grounded data analyses. In the following subsections, the main algorithm steps are described in detail.

4.1.2 Document parsing

The first step is to parse the input Web document (line 3) which is supposed to describe a particular real world entity, from now on *ae*. The *Parse_document* function depends on the kind of document that is being analysed. If it is a HTML document, then it is necessary to extract raw text from it by means of HTML parsers which are able to remove headers, templates, HTML tags, etc. Otherwise, if the document comes from a semi-structured source such as Wikipedia, then ad-hoc tools are used to filter and extract the main text.

4.1.3 Named Entity detection

This step consists in extracting relevant named entities from the analysed document. The function *extract_potential_NEs* (line 6) returns a set of Named Entities (*PNE*). The implementation of this function depends on the type of input, as will be described in section 4.1.5.1. However, as stated above, only a subset of the elements of *PNE* really describe the main features of *ae*. The rest of the elements of *PNE* could introduce noise because they may be unrelated 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 8). To do that, a Web-based co-occurrence measure that tries to assess the degree of relationship between *ae* and each *NE* is used. Concretely, a version of the *Pointwise Mutual Information* (PMI, introduced in Eq. 2 and Eq. 3) relatedness measure adapted to the Web is computed (Church et al., 1991).

$$NE\_SCORE(pne_i, ae) = \frac{hits(pne_i \ \& \ ae)}{hits(pne_i)} \tag{5}$$

In the NE_SCORE (Eq. 5), concept probabilities are approximated by Web hit counts provided by a Web search engine as proposed by (Turney, 2001). Since the intention of the algorithm is to rank a set of choices –PNEs- for a given domain (i.e. the *ae*) the *hits(ae)* term that should appear in the denominator can be dropped

because it has the same value for all choices. NEs that have a score exceeding an empirically determined threshold (*NE_THRESHOLD*, line 8) are considered as relevant, whereas the rest are removed. The value of the threshold determines a compromise between the precision and the recall of the system, as will be shown in the evaluation results presented in section 4.2.

4.1.4 Semantic Annotation

In this work the aim of the semantic annotation step is to match the extractions found in a text describing an entity with the appropriate classes contained in the input ontology, which models the features of interest.

Some approaches have been proposed in this area. 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 them. 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 that try to discover automatically taxonomic relationships (Bisson et al., 2000; Sanderson & Croft, 1999), 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 (Turney, 2001), but this solution is not scalable because of the huge amount of required queries (Cimiano et al., 2005).

The method proposed in this work employs this last technique, but introducing a previous step that reduces the number of queries to be performed. To do so, the semantic annotation step is divided in two parts: the discovery of potential subsumer concepts (line 14) and their matching with the ontology classes (lines 19-38).

4.1.4.1 Discovering potential subsumer concepts

This first stage is proposed in order to minimise the number of queries (NE, ontology class) to be performed by the final statistical assessor. It may be noticed that the problem of semantic annotation is to find 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). As stated in before NEs and concepts are semantically related by means of taxonomic relationships. Thus, the way to go from the instance level to the conceptual level is by discovering taxonomic abstractions, which are represented by subsumer concepts. So, in this stage, the aim is to automatically discover possible subsumer concepts for each NE.

This is done by means of the function *extract_subsumer_concept*s, which depends on the kind of input document (see section 4.1.5.2 for specific details). As a result, we obtain a set of subsumer concepts for each NE (e.g. the subsumer concepts of *Porsche* could be *car*, *automobile*, *auto*, *motorcar* and *machine*). Notice that those concepts are abstractions of the NE and they not depend on any ontology. This means that subsumer concepts do not necessarily match with ontological classes. However, at this point the algorithm is working at a conceptual level (rather than at the instance level) and thus it is possible to match NEs with ontological classes more easily and efficiently in the following step.

4.1.4.2 Matching subsumers to ontological classes

This stage tries to discover a correspondence between the subsumer concepts of a NE and the ontological classes. To do so, two different cases are considered: Direct Matching and Semantic Matching.

4.1.4.2.1 Direct Matching

First, the system tries to find a direct match between the subsumers of a NE and the ontology classes. This phase begins with the extraction of all the classes contained in the domain ontology (line 19). Then, for each Named Entity ne_i, all its potential subsumer concepts ($sc_i \in SC$) are compared against each ontology class in order to discover lexically similar ontological classes ($soc_i \in SOC$, lines 23-25), 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 (Porter, 1997) is applied to both the sc_i and the 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* 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, subsumers do not appear as ontology classes with exactly the same name. As a result, potentially good candidates for annotation are not discovered.

4.1.4.2.2 Semantic Matching

This step faces the non-trivial matching between semantically similar subsumers and ontological classes that, because they were represented with different textual labels, could not be discovered in the previous stage. Details on how this task is performed are given in Algorithm 2, which specifies what is done in line 28 of Algorithm 1.

Algorithm 2. Semantic matching method (line 28, Algorithm 1)

```
1  extract_semantic_matching(OC, SC, ne, ae) {
2    ∀ sc_i ∈ SC {
3          SYNSETS := get_wordnet_synsets (sc_i)
4          if |SYNSETS|==0 {return}
5          else if |SYNSETS|==1 {
6                RELATED_TERMS ← get_related_terms(synset_0)
7          }
8          else { /*when the sc has more than 1 synset, disambiguation is needed*/
9                CONTEXT := get_context(ne)
10               CONTEXT := CONTEXT + get_web_snippets(ne, ae)
11               ∀ context_j ∈ CONTEXT {
12                     /* the similarity between the context
13                     and each synset is calculated */
14                     ∀ synset_k ∈ SYNSET {
15                           similarity := get_cosine_distance(context_j, synset_k)
16                           update_average_synset(synset_k, similarity)
17                     }
18               }
19               disambiguated_synset :=get_synset_max_average(SYNSETS)
20               RELATED_TERMS := get_related_terms(disambiguated_synset)
21         }
22   }
23   return extract_direct_matching(OC, RELATED_TERMS)
24 }
```

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 *SOC*, so that the direct matching can be re-applied 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 the algorithm works at a conceptual level, WordNet (Fellbaum, 1998) has been used to obtain these related terms and to increase the *SOC* set.

The main problem of semantic matching is that a word may be polysemous and, before extracting the related concepts from WordNet, it is necessary to discover to which sense it corresponds (i.e., a semantic disambiguation step must be performed to choose the correct synset). To do so, one possible solution is to use the context (i.e., the sentence from which ne_i was extracted, line 9 of Algorithm 2) but, usually, that is not enough to disambiguate the meaning. To minimise this problem, the Web is used to assist the disambiguation.

We propose a Web-based approach that combines the contexts of Named Entities, WordNet definitions and the cosine distance. The Web is used in this case to find new evidences of the relationship between the *ne* and the *ae* in order to increase the contexts (where those terms occurs together) that enable a better disambiguation of the meaning of polysemous words. First, the algorithm retrieves the contexts in which the NE appears (line 10 of Algorithm 2), framed in the scope of the original *ae* (by querying a search engine for common appearances of *ae* and *ne*). Then, the set of Web snippets returned by the search engine is analysed. Web snippets (see Figure 8) are used instead of complete documents because they

precisely contain, in a concise manner, the context of the occurrence of the Web query. Several snippets (e.g. 10) can be obtained with a single Web query, so their analysis is much more efficient than the one of complete Web resources.

Figure 8 Snippet obtained by Google for the Sagrada Familia NE

Then, the system calculates the cosine distance between each snippet and every WordNet synset of the element of *SC* (lines 11 to 18 of Algorithm 2). The aim is to assess which of the senses of each subsumer concept is the most similar to the context in which the NE appears. The synset with the highest average value is finally selected as the appropriate conceptualization of the subsumer concept (line 19 of Algorithm 2) and the direct matching is repeated with all the new SCs.

Let us consider an example to illustrate this method. Table 8 depicts the input data of the problem. The first three rows are the analysed entity, the named entity and a subsumer concept. The last three rows show the query performed in order to retrieve Web snippets and the two WordNet synsets of the subsumer concept.

Table 8 Semantic disambiguation example (part I)

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 9 shows a subset of the retrieved snippets, which represent contexts of the NE, and the cosine distance between the context and each of the subsumer's synsets. Synset 1 obtains the highest score, so it will be selected as the most appropriate conceptualization of the SC and, hence, its synonyms, hyponyms and hypernyms will be retrieved in the semantic matching step. In this example, *minster*, *church* and *church building* are used as terms to expand the direct matching between the SC *cathedral* and the ontology classes.

Table 9 Semantic disambiguation example (part II)

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
[...]	[...]	[...]
Final Average	**0.14**	**0.12**

4.1.4.3 Class Selection

After applying the matching process between subsumers and ontological classes, a final assessor decides if the selected matching is adequate enough (if only one matching was found) or which of the matchings is the most appropriate, if several ones were obtained (line 32 of Algorithm 1). In this case, a Web-based statistical measure is applied again in order to choose the most representative one (Class Selection, lines 33-34 of Algorithm 1). Notice that, at this stage, the number of matches between NEs and ontological classes is low compared with all the possible combinations between NEs and ontological classes (as discussed previously). As a result, a much lower amount of queries is needed to select the final annotation. This shows the benefits of exploiting automatically discovered subsumer concepts as a bridge to enable more direct semantic annotations.

The selection is based on the degree of relatedness between the Named Entity and each element of the matched ontological classes in *SOC*, assessed again with the Web-based version of PMI introduced in section 2.3.3. However, it must be noted that the elements of *SOC* can also be polysemous, and can be referring to different concepts depending on the context (line 33 of Algorithm 1). So, in Eq. 5, the analysed entity *ae* has been introduced to contextualise the relationship of each element of *SOC* with ne_i.

$$SOC\_Score(soc_i, ne_i, ae) = \frac{hits(ae \,\&\, ne_i \,\&\, soc_i)}{hits(ae \,\&\, soc_i)} \quad (6)$$

The score (Eq. 6) computes the probability of the co-occurrence of the named entity ne_i and each ontology class proposed for annotation soc_i from the Web hit count provided by a search engine when querying these two terms (contextualised with *ae*). Finally, the annotation with the highest score which exceeds the AC_THRESHOLD (lines 34-35 of Algorithm 1) is annotated. If no elements in SOC exceed the threshold, the NE remains unannotated. This will indicate that, even though the NE and its corresponding SC may be correct, there are not enough evidences (statistically assessed from the Web) to support the annotation in the context of the domain defined by the input ontology.

4.1.5 Extraction of features from raw texts

So far, the generic feature extraction algorithm has been presented. This section discusses how the functions *extract_potential_NEs* (line 6 of Algorithm 1) and *extract_subsumer_concepts* (line 14 of Algorithm 1) have been implemented in order to apply it to raw text.

4.1.5.1 Named Entities detection

The main problem related with the detection of NEs from raw text is the fact that they are unstructured and unlimited by nature (Sánchez, Isern, & Millan, 2011). This implies that, in most cases, NEs are not contained in classical repositories like WordNet due to its potential size and its dynamic character. Different approaches have been proposed in the field of NE detection. Roughly, they can be divided into supervised and unsupervised methods.

Supervised approaches rely on a specific set of extraction rules learned from pre-tagged examples (Fleischman & Hovy, 2002; Stevenson & Gaizauskas, 2000), or predefined knowledge bases such as lexicons and gazetteers (Mikheev & Finch, 1997). However, the 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), hampering the recall (Pasca, 2004).

In *unsupervised* approaches like (Lamparter, Ehrig, & Tempich, 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 coverage 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 a 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 (Cimiano et al., 2004; Downey, Broadhead, & Etzioni, 2007; Pasca, 2004), provides good results without depending on manually annotated examples or specific categories.

We have defined the following unsupervised, domain-independent method to detect NEs when dealing with raw text (i.e., to implement *extract_potential_NEs*, line 6). First, a natural language processing parser is used. Concretely, the four modules of the OPENNLP[5] parser (Sentence Detector, Tokenizer, Tagging and Chunking) are applied in order to analyse syntactically the input text of the Web document. Then, all the detected Noun Phrases (NP) which contain one or more words beginning with a capital letter are considered as a Potential Named Entities (*PNE)*.

Table 10 shows an example of the extracted PNEs from the first fragment of text of the Wikipedia article about Tarragona[6].

Table 10 Set of extracted NE from Tarragona Wikipedia introduction

Detected sentences	Extracted PNE	Correct?
[NP Tarragona/EX]	Tarragona	yes
[NP Catalonia/NN]	Catalonia	yes
[NP Spain/NNP]	Spain	yes
[NP Sea/NNP]	Sea	no
[NP Tarragonès/VBZ]	Tarragonès	yes
[NP the/VBZ Vegueria/NNPS]	the Vegueria	no

4.1.5.2 Discovering potential subsumer concepts

To extract subsumer concepts of NEs from raw text, it is necessary to look for linguistic evidences stating their implicit taxonomical relationship. As discussed in section 2.3.2, we use Hearst's taxonomic linguistic patterns, which have proved their effectiveness to retrieve hyponym/hypernym relationships (Hearst, 1992). The Web is used to assist this learning process, looking for pattern matches that provide the required linguistic evidences.

To do so, the system constructs a Web query for each NE and each pattern. Each query is sent to a Web search engine, which returns as a result a set of Web

[5] http://incubator.apache.org/opennlp/ Last access: July 24th, 2012

[6] http://en.wikipedia.org/wiki/Tarragona. Last access: November 10th, 2014

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

Table 11 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

4.1.6 Extracting features from semi-structured documents

In this work, Wikipedia has been used to show the applicability of the proposed method when applied to semi-structured resources. Wikipedia has some particularities which can ease the information extraction when compared with raw text. This work focuses on the exploitation of *internal links* and *category links*. The first ones represent connections between terms that appear in a Wikipedia article and other articles that describe them. This is an indication of the fact that the linked term represents a distinguished entity by itself, easing the detection of NEs. On the other hand, *category links* group different articles (corresponding to entities) in areas that are related in some way and give articles a kind of categorization. Wikipedia's category system can be thought of as consisting of overlapping trees. Hence, Wikipedia categories can be considered as rough taxonomical abstractions of entities and can be exploited to aid the annotation process.

This section details how the functions *extract_potential_NEs* (algorithm 1, line 6) and *extract_subsumer_concepts* (algorithm 1, line 14) have been adapted to take advantage of these semi-structured data.

4.1.6.1 Named Entities detection

In order to take profit of the manually created Wikipedia link structure, those words tagged with internal links have been considered as potential named entities (*PNE*). In this way the analysis is reduced to only those terms that are likely to correspond to distinguished entities (that have its own Wikipedia article).

The problem of the *PNEs* extracted from the internal links are that, on the 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 NEs.

In order to illustrate these problems, the following fragment of text extracted from the Wikipedia article corresponding to the city of Barcelona is 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 internally linked with other Wikipedia articles. Three of them are NEs (*Catalonia*, *Spain* and *Madrid*) and they represent instances of cities/regions/countries, whereas the other one is a common noun which represents a concept (*capital*). Moreover, 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 potential problems, the set of extracted *PNEs* (see Table 12 for an example of some of the options forund for Barcelona) has to be filtered by means of the NE score presented in the general algorithm (see section 4.1.3), as it is done with the *PNEs* obtained from the analysis of raw text. In this case, however, the algorithm starts with a pre-processed set of NEs whose reliability is supported by the community of Wikipedia contributors.

Table 12 Subset of extracted NEs from Barcelona Wikipedia article

Wikilinks	Correct?
Acre	no
Antoni Gaudí	yes
Arc de Triomf	yes
Archeology Museum of Catalonia	yes
Barcelona Cathedral	yes
Barcelona Museum of Contemporary art	yes
Barcelona Pavilion	yes
Casa Batlló	yes

4.1.6.2 Discovering potential subsumer concepts

In order to extract potential subsumer concepts for each named entity, Wikipedia category links have been used. The idea is to consider Wikipedia

categories of articles describing a NE as potential subsumers of the NE. In this way it is possible to obtain, in a very direct and efficient manner, the set of subsumers that enable the matching of the NE with ontological classes. In Table 13 there are examples of potential subsumer concepts of the Nes shown in the previous table.

Table 13 Subset of extracted potential subsumer concepts for Barcelona NEs

Articles	Potential subsumer concepts
Antoni Gaudí	1852 births, 1926 deaths, architects, roman catholic churches, art nouveau architects, catalan architects, spanish ecclesiastical architects, modernisme architects, 19th century architects, 20th century archi tects, organic architecture, people, reus...
Arc de Triomf	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…
Casa Batlló	visionary environments, antoni gaudí buildings, 1907 architecture, world heritage sites, spain, visitor attractions, eixample, passeig, gràcia, outsider art, 1907 works, 1900s architecture, edwardian architecture...

A problem of these categories, however, is the fact that they are, in many cases, too complex or concrete to be modelled in an ontology. For example, *The Sagrada Familia* article is categorised 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. These categories are too complex because they involve several concepts or even mix concepts with other NEs. To tackle this problem, we syntactically analyse each category to detect the basic concepts to which it refers. 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 the Wikipedia categories is the fact that they do not always contain enough concepts to perform the matching among them and

ontological classes, or those are too concrete to be included in an ontology (and, hence, to enable a direct matching). Fortunately, as mentioned before, Wikipedia categories are included in higher categories. So, we consider two upper-level categories to improve the recall of the ontology matching. As the Wikipedia categorisation does not define a strict taxonomy, it is not advisable to climb recursively in the category graph because the initial meaning could be lost.

4.1.7 Computational cost

The execution time of the proposed method mainly depends on the number of queries, since the Web search engine response time is several orders of magnitude higher (around 500 milliseconds) than any other offline analysis performed by our method (e.g. natural language processing, pattern matching, ontological processing, etc. only take a few milliseconds). There are five different steps in which queries are performed: NE detection, NE filtering, subsumer concepts extraction, semantic disambiguation and class selection. Since the response time of the Web search engine does not depend on the type and complexity of the query (but on the delay introduced by the Internet connection), the execution time for all kinds of queries is expected to be equivalent.

Both plain text and semi-structured text analyses have the same cost for NE filtering, semantic disambiguation and class selection. To rank NEs in the relevance filtering step, two queries are required by the *NE_Score* function (4) to evaluate each NE. Hence, a total of $2n$ queries are performed, where n represents the number of NEs. Class selection requires computing as many SOC_Scores (5) as candidates. Hence, being c the total number of class candidates, this results in $2c$ queries. In the semantic disambiguation stage, only one query is needed for each candidate to obtain the snippets relating each candidate with the analysed entity (i.e. c queries are performed in this step).

Hence, the difference in computational cost between plain text analyses and semi-structured ones is in the steps of NE detection and extraction of subsumer concepts. Although NE detection is different when analysing plain texts and semi-structured resources, neither of them needs to perform queries and its cost is considered constant for both approaches. Referring to the extraction of subsumer concepts, in the first approach six queries are performed to discover subsumer concepts by means of Hearst Patterns ($6n$), whereas in the second approach no queries are needed because SCs are directly extracted from the categories of the tagged entities.

In summary, considering the above expressions, the number of queries needed to analyse plain text is $2n+6n+2c+1c=8n+3c$, whereas only $2n+2c+1c=2n+3c$ are needed when dealing with Wikipedia articles. In both cases the proposed method scales linearly with respect to the number of NEs, but the much lower coefficient for the Wikipedia articles shows how the exploitation of their structure aids to improve the performance of the method.

4.2 Evaluation

This section presents the evaluation of the proposed method, which has been conducted in three directions. In the first part, given a specific evaluation scenario, we present a detailed picture of the influence of the different parameters involved in the learning process: thresholds, input ontology and document types. In the second part, in order to give a general view of the expected results, a collection of tests for two different scenarios (i.e. different input documents and ontological domains) is presented. The last part provides an *extrinsic* evaluation of the feature extraction algorithm, which shows the usefulness of its results in the construction of a personalised recommender of Tourism activities.

In the first two cases, the results have been evaluated according to their precision and recall against ideal results (provided by a human expert). To do so, the expert has been requested, for each input document representing an analysed entity (*ae*) and ontology, to manually select which features found in the document refer directly or indirectly to concepts modelled in the ontology. As a result, a list of ideal features corresponding to concepts found in the ontology for each *ae* is obtained.

Then, recall is calculated by dividing the number of correct features discovered by our system (i.e. those also found in the ideal set of features) by the amount of ideal features stated by the human expert.

$$RECALL = \frac{\#Correct\_features}{\#Ideal\_features} \tag{7}$$

Precision is computed as the number of correct features (as above) divided by the total number of features retrieved by our system.

$$PRECISION = \frac{\#Correct\_features}{\#Retrieved\_features} \tag{8}$$

In the last case the measures of precision and recall of the semantic recommender system have been computed with regards to an ideal list of recommendations, which would be obtained if a costly computational comparison between the characteristics of the destinations and the preferences of the user was performed

4.2.1 Influence of input parameters

In this first battery of tests we focused on documents describing cities (Barcelona and Canterbury), using ontologies related to touristic and geo-spatial features. First, the influence of the algorithm thresholds in the results has been studied. They enable to configure the system behaviour so that a high precision, a high recall or a balance between both of them can be achieved. Then, different types of

input documents have been considered. Specifically, our method has been applied to Wikipedia articles, but analysing them either as plain text or as Wiki-tagged documents. Finally, different input ontologies covering the same domain of knowledge have been used to evaluate the influence of the ontology design and coverage in the results. Two ontologies have been considered: a manually built Tourism ontology[7], which models concepts related to different kinds of touristic points of interest typically found in Wikipedia articles, and a general ontology modelling geo-spatial concepts (Space[8]) retrieved from the SWOOGLE[9] search engine. A summary of their structure is shown in Table 14

Table 14 Description of used ontologies

Ontology	Taxonomical depth	#classes	Root classes
Tourism	5 levels	315	Administrative divisions, buildings, festivals, landmarks, museums, sports
Space	6 levels	188	Geographical features, geopolitical entities, places

4.2.1.1 Learning thresholds

In this section, the influence of the thresholds used to filter named entities (NE_THRESHOLD) and to select annotations (AC_THRESHOLD) is studied. The Wikipedia article of Barcelona, taken as plain text and as a Wiki-tagged document, and the general Space ontology have been used as input. Figure 9 shows the precision and recall figures when setting one of the two thresholds to 0 and varying the other one from 0 to 1 for the different types of input documents.

[7] http://deim.urv.cat/~itaka/CMS2/images/ontologies/tourismowl.owl Last access: November 10th, 2014

[8] http://deim.urv.cat/~itaka/CMS2/images/ontologies/space.owl Last access: November 10th, 2014

[9] http://swoogle.umbc.edu/ Last Access: July 24th, 2012

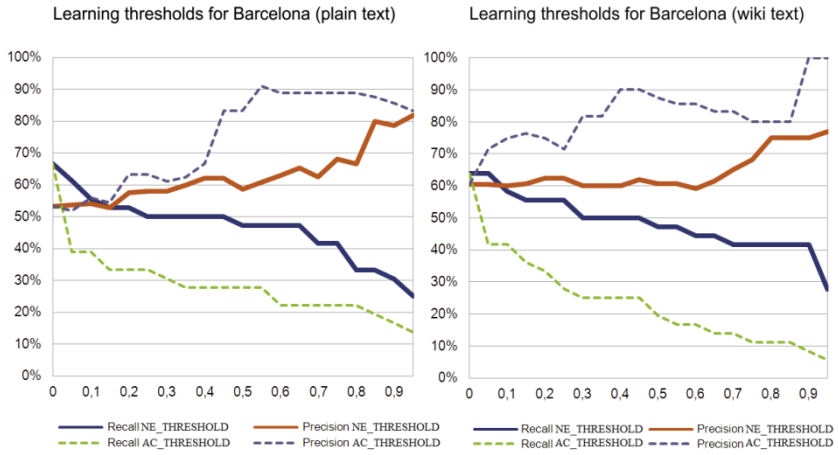


Figure 9. Influence of NE_THRESHOLD and AC_THRESHOLD

Results show that the AC_THRESHOLD has a larger influence in precision/recall. Notice that the NE_THRESHOLD is related to the NE_Score, which is calculated by measuring the level of relatedness between the potential named entity and the analysed entity, whereas AC_THRESHOLD goes further by measuring the relatedness between the analysed entity, the potential named entity and the subsumer candidate to be annotated (SOC_Score). This fact implies that the second threshold is more restrictive because the relatedness involves three elements instead of two. Moreover, since the AC_THRESHOLD is considered after the NE_THRESHOLD, its purpose is twofold: 1) it measures the relatedness between the named entity and its subsumer candidate, facilitating the final annotation, and 2) it contextualises the ontology annotation in the domain of the analysed entity, so that it may filter unwanted named entities. This later aspect is similar to the purpose of the NE_THRESHOLD, which is to select only those NEs that are related to the analysed entity. Since all NEs selected according to the NE_THRESHOLD have to pass a second more restrictive threshold, this enables the system to drop irrelevant entities and those that are not related with concepts modelled in the input ontology. However, the NE_THRESHOLD is useful to drop some NE candidates to be analysed in latter stages and, hence, to lower the number of required Web queries associated to the evaluation of their subsumer concept candidates.

From a general perspective we observe an inverse behaviour for precision and recall and for both thresholds. Since the final goal of our method is to enable the application of data analysis methods (such as clustering) a high precision may be desirable, even at the cost of a reduced recall. In this case, more restrictive thresholds can be used. This aspect shows an important difference with related works in which no statistical assessor is used to filter extractions (Cimiano et al., 2005). Since no parameter tuning is possible, the precision of the final results may be limited by the fixed criterion for NE selection. Notice also that, analysing the document as plain text, a total of 202 named entities were detected from which

only 146 were already annotated by Wikipedia (i.e. 72%). So, in comparison with methods like the one proposed by (Zavitsanos, Tsatsaronis, Varlamis, & Paliouras, 2010), which limit the extraction to what it is already annotated in input knowledge bases (i.e. WordNet, Wikipedia, etc.), our method is potentially able to improve the extraction and annotation recall.

4.2.1.2 Plain text vs. Wiki-tagged document

In this second test, we picked up as case studies the Wikipedia articles that describe the cities of Barcelona and Canterbury, analysing them as plain text and as wiki-tagged documents. In all cases, the Space ontology was used. Again, different threshold values were set as shown in Figure 10.

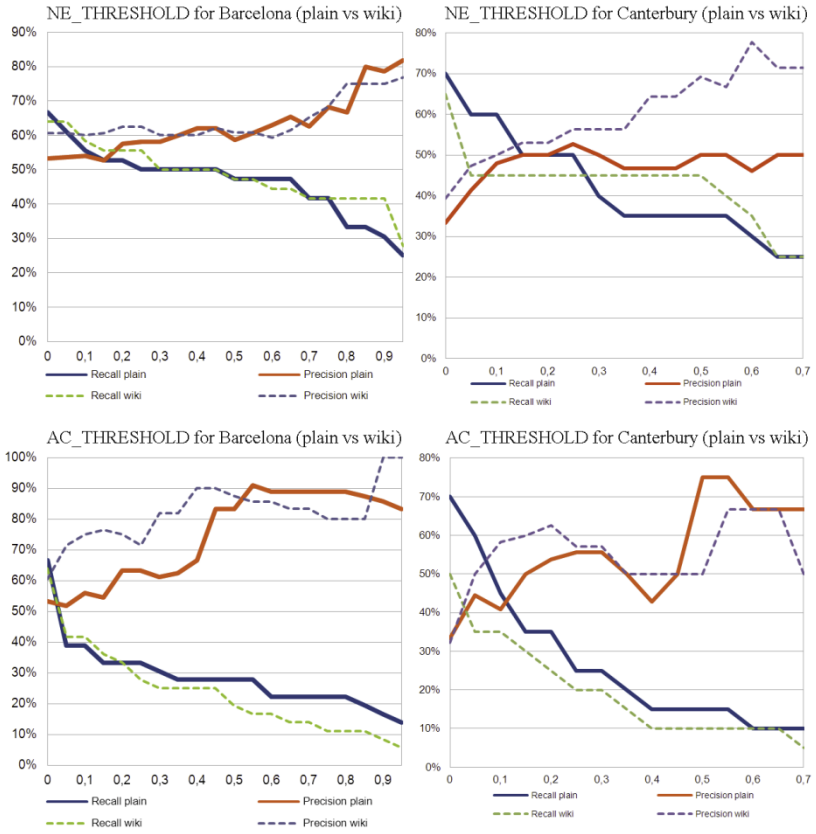


Figure 10 Plain text vs. Wikipedia documents

Figures show that precision tends to remain stable or to improve when taking into account the additional information provided by Wiki-tagged text, that is, the tagged entities and their associated Wiki-categories. On the contrary, recall is, in

many situations, higher when analysing the input document as plain text. This is because when using plain text, the whole textual content is analysed. Hence, there are more possibilities to extract named entities that could correspond to representative features than when relying solely on Wiki-tagged entities. On the contrary, since Wiki entities are tagged by humans, the precision of their resulting annotations is expected to be higher than when performing the automatic NE detection proposed by our method. Moreover, for the Wiki-tagged text, humanly edited categories are used to assist the annotation, which may also contribute to provide more accurate results. In any case, results are close enough to consider the plain text analysis (i.e. both the NE detection and the subsumer candidate learning) almost as reliable as human annotations, at least for feature discovery purposes. It is important to note, however, that the analysis of Wiki-tagged text, as discussed previously, is considerably faster than its plain text counterpart, since the number of analyses and queries required to extract and annotate entities is quite lower.

4.2.1.3 Input ontologies

The third test compares the results obtained using different ontologies to assist the feature extraction process of a given entity. In this case, the plain-text and wiki-tagged versions of the Barcelona article have been analysed using the input ontologies Space and Tourism. Again, results for different threshold values are shown in Figure 11.

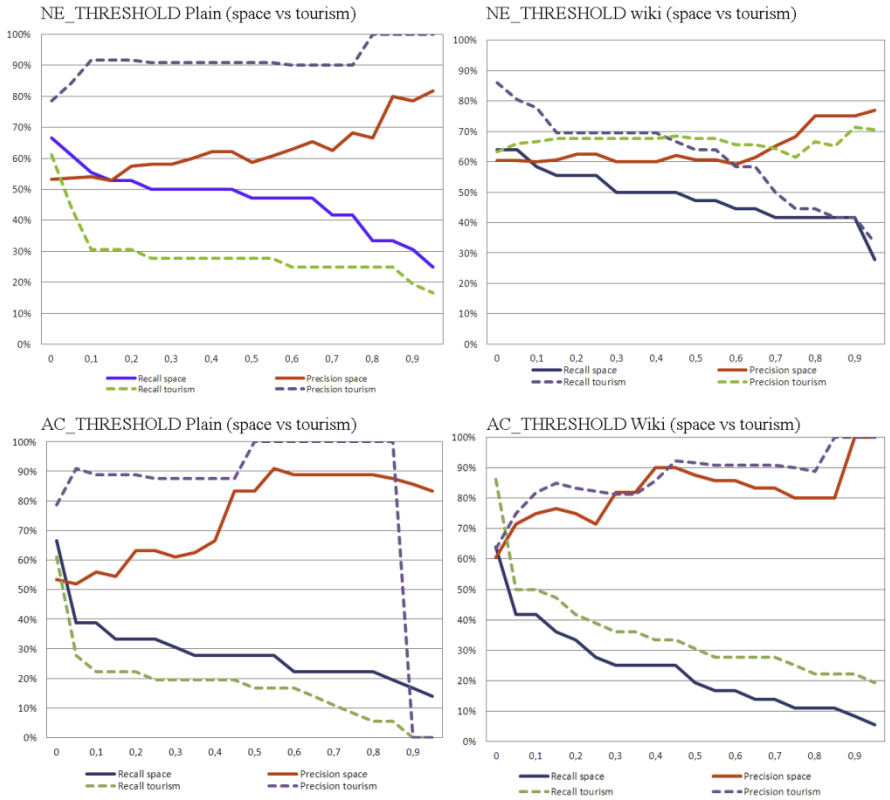


Figure 11 Influence of domain ontologies

Results obtained for the plain-text version of the Barcelona article show a noticeably higher precision when using the Tourism ontology instead of the Space one. As the former ontology was constructed according to concepts typically appearing in Wikipedia articles describing cities, it helps to provide more appropriate annotations since, even though extracted named entities are equal for both ontologies, the Tourism one is more suited to annotate those that are already tagged in the Wikipedia article. Moreover, since the Tourism ontology contains more concrete concepts than the more general-purpose Space one, language ambiguity that may affect the subsumer extraction stage and the posterior annotation is minimised, resulting in better precision. The downside is the lowered recall obtained with the Tourism ontology, since solely those named entities that are already tagged in the text (and that, hence, correspond to concepts modelled in the Tourism ontology) are likely to be annotated. Differences between the two ontologies when using Wiki-tagged text are much reduced. In this case, the fact that NEs correspond solely to those already tagged in the Wikipedia article produces more similar annotation accuracies. Recall is, however, significantly higher for the Tourism ontology, since NE subsumers, which correspond to Wikipedia categories in this case, are more likely to match directly (as described

in section 4.1.4.2.1) to concepts of an ontology that is specially designed to cover topics and entities usually referred to in Wikipedia documents. This avoids the inherent ambiguity of the semantic matching process (described in section 4.1.4.2.2) that is carried out when a direct matching is not found, which may hamper the results.

4.2.2 Global performance

In this second part of the evaluation, general tests have been performed for several entities belonging to two well differenced domains: touristic destinations and movies. For each domain, eight different entities/documents have been analysed, using their corresponding Wikipedia articles. For touristic destinations, the Tourism ontology introduced above has been used, whereas for movies, a film[10]ontology retrieved from SWOOGLE has been used. This latter one models concepts related with the whole production process of a film, including directors, producers, main actors/actresses starring in the film, etc. The structure of both ontologies is summarised in Table 15.

Table 15 Description of used ontologies

Ontology	Taxonomic depth	#classes	Main classes
Tourism	5 levels	315	Administrative divisions, buildings, festivals, landmarks, museums , sports
Film	4 levels	20	Person, product, company

Table 16 depicts the precision and recall obtained for the different cities analysed in the tourism domain, whereas Table 17 shows the results of the evaluation of films.

Table 16 Evaluation results from Wikipedia descriptions of different cities using *Tourism* ontology

Articles	Precision	Recall
Cracow	80%	36.3%
London	75.8%	66.6%
New York City	69.2%	75%
Poznan	66.6%	44.4%
Saint Petersburg	66.6%	57.1%
Stockholm	73.6%	56%
Vancouver	45.4%	50%
Warsaw	79.1%	70.3%
Average	**69,5%**	**57,0%**

[10] http://deim.urv.cat/~itaka/CMS2/images/ontologies/film.owl Last access: November 10th, 2014

Table 17 Evaluation results from Wikipedia descriptions of different films using *Film* ontology

Articles	Precision	Recall
Batman Begins	75%	52%
First Blood (Rambo)	87.5%	77.7%
Inception	81.2%	72.2%
Matrix	87.5%	57%
Oldboy	57.1%	66.6%
Pulp Fiction	78%	61.1%
Requiem for a dream	60%	50%
The Fountain	88.8%	69.2%
Average	**76.9%**	**63.2%**

Comparing both domains, we observe slightly better results for the film domain, with averaged precision and recall figures of 76,9% and 63,2%, respectively. This is related to the lower ambiguity inherent to film-related entities, in comparison with touristic ones. For example, in the film domain, most NEs refer to person names (directors, producers, stars), which are hardly ambiguous due to their concreteness. Hence, annotation to ontological concepts is usually correct. On the contrary, a NE such as "Barcelona" may refer either to a city or a football team, being both annotations almost equally representative; if both concepts are found in the ontology, the annotation is more likely to fail, in this case.

Analysing individual entities, we observe better precisions and recalls of English/American entities (e.g, American films like First Blood or British cities like London), which usually result in a larger amount of available Web resources for the corresponding entities and, hence, on more robust statistics.

As a general conclusion, considering recall figures, we observe that our method was able to extract, in both cases, more than half of the features manually marked by the domain expert. Moreover, precisions were around a 70-75% in average, which produce usable results for further data analysis. It is important to note that these results, which have been automatically obtained, suppose a considerable reduction of the human effort required to annotate textual resources.

4.2.3 Semantic recommendation of touristic destinations

In the context of the DAMASK research project an extrinsic evaluation of the results of the feature extraction algorithm presented in this chapter was made, by using them to improve the quality of the suggestions on Tourism destinations provided by a recommender system.

The main tasks undertaken in the DAMASK (*Data Mining Algorithms with Semantic* Knowledge) research project were the definition of new ontology-based Information Extraction tools from heterogeneous resources (C Vicient, Sanchez,

& Moreno, 2011a, 2011b; C Vicient, Sánchez, & Moreno, 2012), the definition of a new semantic similarity measure between lists of ontological concepts (Moreno et al., 2014; Moreno, Valls, Mata, et al., 2013), the adaptation of classical clustering algorithms so that they could deal with numerical, categorical and multi-valued semantic attributes ((Moreno, Valls, Mata, et al., 2013)) and the test of all those new methods in the development of a prototype of a personalised recommender of holiday destinations. This section focuses on this latter task. First, more details concerning the construction of the data matrix and the clustering process are given. After that, a detailed quantitative analysis of the accuracy of the recommender system on four diverse case studies is provided.

4.2.3.1 Information extraction and clustering

One of the first steps of the project was the manual construction of a comprehensive Tourism ontology that contains 538 classes connected in 9 hierarchy levels. It is structured around 4 main concepts that constitute the first level of the hierarchy: "*Geopolitical Division*", "*Activity*", "*Point of Interest*" and "*Geographical Feature*". The ontology has multi-inheritance between concepts.

The prototype of the recommender system considered the 150 leading and most dynamic cities in terms of tourist arrivals, according to the ranking made by Euro Monitor International in 2006 (Bremmer, 2007). This wide set covers cities all around the world. A set of 2 numerical, 2 categorical and 8 multi-valued semantic attributes was defined to represent the information about the cities. The automatic Information Extraction methods explained in this chapter, which were defined in the first phase of the DAMASK project, were applied on the Wikipedia pages of the 150 cities to gather, in an unsupervised fashion, the information about the values of the attributes. The presence of a semantic value (e.g. the religious buildings in the city) required the textual analysis of the content of the Wikipedia page (C Vicient et al., 2012).

In the tests reported in this section the following 8 multi-valued semantic attributes related to leisure activities and tourist places were considered:

- Aquatic and Nature sports: swimming, rafting, diving, climbing…
- Other sports: motor, martial, golf, dance, tennis, cricket, football...
- Religious buildings: synagogue, mosque, chapel, temple, cathedral…
- Cultural buildings: school, opera, forum, university, library, theatre…
- Other interesting buildings: skyscraper, casino, stadium, mall, palace...
- Museums: modern art, Egyptian, toy, technology, Natural History…
- Landmarks related to Water and Geography: hill, bridge, canal, lake…
- Other Landmarks: park, memorial, tomb, forest park, fountain...

Cities may have a missing value in some semantic attributes, showing that the IE procedures have not found any evidence for the possible values of that attribute. It has to be taken into account that, as these procedures are unsupervised and automatic, the quality of the data matrix construction is slanted by the *precision* and *recall* of each of the steps of the IE process (the natural language parser, the named entity detection and the Web statistics used to estimate the similarity between the retrieved terms and the ontology concepts, as explained in this chapter). A maximal distance of 1 (for a specific attribute) was set for the case of comparing a city with data and a city with a missing value. In the case of having two cities without values, this atribute is not taken into account in the computation of the distance between them; thus, only attributes for which at least one of the cities contains information are considered.

The goal in the final task of the project was to make a personalised recommendation of tourist destinations, taking into account the preferences of the user. The cities considered by the system were grouped using the adapted k-means algorithm for which a new procedure for managing the semantic attributes was defined (including the definition of a multi-valued centroid and the multi-valued semantic distance based on the Tourist ontology, (Moreno, Valls, Mata, et al., 2013)). The preferences stated by the user were then compared with the centroids of these clusters, and the user was recommended the cities belonging to the most similar cluster(s). The hypothesis, which is proven to be correct in the next subsection, is that the pre-clustering of the destinations allows a strong reduction on the computational cost of the recommendation process, as it is not necessary to compute the resemblance of each destination with respect to the preferences of the user, without reducing the quality and accuracy of the recommendations.

4.2.3.2 Study of the accuracy of recommendations

In this section the recommendations provided by the system are compared with those that would be made without the previous clustering, comparing directly the user preferences with the whole list of cities. Four user profiles have been defined to test the recommendations in different conditions (from a specific profile with very concrete requirements satisfied by a low number of cities to a very general one with wide requirements fulfilled by most of the cities). In order to numerically quantify the quality of the recommendations, the F1 score, that considers both the precision and the recall of the test, is computed. In this manner it is possible to objectively quantify the similarity between the ideal recommendations made directly from the list of cities and the ones made by the system.

A profile is described by its preferences regarding the 8 multi-valued semantic attributes previously commented. Table 18 shows the four different profiles that have been considered in the tests and the preferences for each attribute

Table 18 Results of the test with 4 different profiles

Id	Aquatic nature sports	Other sports	Religious buildings	Other buildings	Museum	Geo-graphical	Other landmark	Cultural building
1	--	--	--	Golf course Fort	--	--	--	--
2	Swimming	Martial art	--	Kiosk Headquarter	Military museum	--	--	Music school
3	Cycling	Ice hockey	--	Golf course	--	--	Botanical garden	Music school
4	Sailing	Football	Church	House	Maritime museum	Square	Park	University

Analysing the table of profile preferences, it can be seen that user 1 has selected only two values in a single attribute (he is interested in cities with forts and golf courses); thus, it is a user with very concrete requirements that only a small number of cities satisfy. User 2 provides 1 value of interest in 4 attributes and 2 values in the "*Other Buildings*" attribute. Profile 3 also has a value of interest in 5 of the 8 attributes. The fourth profile is the most general one. Not only it provides interests on all the attributes, but they are very generic and easy to find in most of the cities (e.g. *Football*, *Church*, *Square*, *Park*, *University*).

The validation of the proposal has been made comparing the results obtained by the system against an *ideal* recommendation. This recommendation can be calculated by sorting, in ascending order, the whole list of cities in function of its distance to the profile. This ordered list is considered as the ranking of cities according to this profile. The closest city to the profile (the first city in the ranking) should be the first recommendation. In order to test the behaviour of the system, a number of scenarios with different numbers of cities to be recommended (5, 10, 15 20, 25 and 30) have been considered. The distance function used to order the cities with respect to a profile quantifies the distance between cities using multi-valued semantic attributes (Moreno et al., 2014; Moreno, Valls, Mata, et al., 2013). Notice that, since a profile is defined with the same attributes as a city, a profile may be considered as a city and used with this distance function.

In the recommender system, a pre-processing step applies the adapted k-means algorithm and classifies 150 cities in 10 classes of different sizes. After that, the user profile is compared with the 10 centroids to find out which of them are the most similar ones to the preferences of the user. The final recommendation is made by selecting, from the best (1, 2 or 3) cluster(s), the closest cities to the profile. The final number of recommended cities is determined by their distance to the profile. Different relative distances to determine if a city should be recommended were evaluated (from 0.1 to 0.5). These distances are normalised with respect to the maximum distance between the profile and the cities of the

clusters selected (which can be 1, 2 or 3 clusters). A city is recommended if its normalised distance to the profile is lower than the specified threshold, which means that it is similar enough to the user's profile.

The accuracy of the recommendations can be computed by comparing them with the ideal recommendation previously described. To do so, the *precision*, the *recall* and the *F1* scores of the recommender system were computed. The ideal recommendations would have a perfect precision and recall, but they would require a lengthy comparison between the user preferences and the characteristics of each of the 150 cities. In more detail, the *precision* of the recommender is the ratio between the number of correct recommendations (those that appear in the *n* first positions of the list, if the aim is receiving *n* recommendations) and the number of recommended cities. The *recall* of the system is the number of correctly recommended cities divided by the total number of cities which the system should have recommended. *F1* is the harmonic mean of precision and recall.

As an example, following tables (Table 19,

Table 20 and Table 21) show the number of recommendations made by our system to user 4. For each ranking of recommendations, the tables show the corresponding precision, recall and F1 values for different intra-cluster relative distances. The F1 values over 70% are highlighted in the tables.

Table 19 Number of recommended cities (Rec.) made by our system to user 4. The table also shows the precision (P), recall (R), F1 and total number of cities recommended (#tcr) compared with the total number of cities in the ideal recommendation (#tcir) for different distance values used (Dist.). In this test, only 1 cluster has been taken into account for the recommendation.

Clusters:1	**Dist:**	**0,1**	**#tcr**	**5**	**Dist:**	**0,2**	**#tcr**	**12**	**Dist:**	**0,3**	**#tcr**	**19**
#tcir	Rec.	P	R	F1	Rec.	P	R	F1	Rec.	P	R	F1
5	1	0,2	0,20	0,20	1	0,08	0,20	0,12	1	0,05	0,20	0,08
10	2	0,4	0,20	0,27	2	0,17	0,20	0,18	2	0,11	0,20	0,14
15	4	0,8	0,27	0,40	4	0,33	0,27	0,30	4	0,21	0,27	0,24
20	5	1,0	0,25	0,40	5	0,42	0,25	0,31	5	0,26	0,25	0,26
25	5	1,0	0,20	0,33	5	0,42	0,20	0,27	5	0,26	0,20	0,23
30	5	1,0	0,17	0,29	7	0,58	0,23	0,33	7	0,37	0,23	0,29
	Dist:	**0,4**	**#tcr**	**22**	**Dist:**	**0,5**	**#tcr**	**22**				
	Rec.	P	R	F1	Rec.	P	R	F1				
5	1	0,05	0,20	0,07	1	0,05	0,20	0,07				
10	2	0,09	0,20	0,13	2	0,09	0,20	0,13				
15	4	0,18	0,27	0,22	4	0,18	0,27	0,22				
20	5	0,23	0,25	0,24	5	0,23	0,25	0,24				
25	5	0,23	0,20	0,21	5	0,23	0,20	0,21				
30	7	0,32	0,23	0,27	7	0,32	0,23	0,27				

Table 20 Number of recommended cities (Rec.) made by our system to user 4. The table also shows the precision (P), recall (R), F1 and total number of cities recommended (#tcr) compared with the total number of cities in the ideal recommendation (#tcir) for different distance values used (Dist.). In this test, 2 clusters have been taken into account for the recommendation.

Clusters:2	**Dist:**	**0,1**	**#tcr**	**8**	**Dist:**	**0,2**	**#tcr**	**18**	**Dist:**	**0,3**	**#tcr**	**33**
#tcir	Rec.	P	R	F1	Rec.	P	R	F1	Rec.	P	R	F1
5	4	0,50	0,80	0,62	4	0,22	0,80	0,35	4	0,12	0,80	0,21
10	8	1,00	0,80	**0,89**	8	0,44	0,80	0,57	8	0,24	0,80	0,37
15	8	1,00	0,53	**0,70**	12	0,67	0,80	**0,73**	12	0,36	0,80	0,50
20	8	1,00	0,40	0,57	16	0,89	0,80	**0,84**	16	0,48	0,80	0,60
25	8	1,00	0,32	0,48	18	1,00	0,72	**0,84**	18	0,55	0,72	0,62
30	8	1,00	0,27	0,42	18	1,00	0,60	**0,75**	21	0,64	0,70	0,67
	Dist:	**0,4**	**#tcr**	**51**	**Dist:**	**0,5**	**#tcr**	**62**				
	Rec.	P	R	F1	Rec.	P	R	F1				
5	4	0,08	0,80	0,14	4	0,06	0,80	0,12				
10	8	0,16	0,80	0,26	8	0,13	0,80	0,22				
15	12	0,24	0,80	0,36	12	0,19	0,80	0,31				
20	16	0,31	0,80	0,45	16	0,26	0,80	0,39				
25	18	0,35	0,72	0,47	18	0,29	0,72	0,41				
30	21	0,41	0,70	0,52	21	0,34	0,70	0,46				

Table 21 Number of recommended cities (Rec.) made by our system to user 4. The table also shows the precision (P), recall (R), F1 and total number of cities recommended (#tcr) compared with the total number of cities in the ideal recommendation (#tcir) for different distance values used (Dist.). In this test, 3 clusters have been taken into account for the recommendation.

Clusters:3	**Dist:**	**0,1**	**#tcr**	**9**	**Dist:**	**0,2**	**#tcr**	**20**	**Dist:**	**0,3**	**#tcr**	**36**
#tcir	Rec.	P	R	F1	Rec.	P	R	F1	Rec.	P	R	F1
5	4	0,44	0,80	0,57	4	0,20	0,80	0,32	4	0,11	0,80	0,20
10	9	1,00	0,90	**0,95**	9	0,45	0,90	0,60	9	0,25	0,90	0,39
15	9	1,00	0,60	**0,75**	13	0,65	0,87	**0,74**	13	0,36	0,87	0,51
20	9	1,00	0,45	0,62	18	0,90	0,90	**0,90**	18	0,50	0,90	0,64
25	9	1,00	0,36	0,53	20	1,00	0,80	**0,89**	21	0,58	0,84	0,69
30	9	1,00	0,30	0,46	20	1,00	0,67	**0,80**	24	0,67	0,80	**0,73**
	Dist:	**0,4**	**#tcr**	**54**	**Dist:**	**0,5**	**#tcr**	**66**				
	Rec.	P	R	F1	Rec.	P	R	F1				
5	4	0,07	0,80	0,14	4	0,06	0,80	0,11				
10	9	0,17	0,90	0,28	9	0,14	0,90	0,24				
15	13	0,24	0,87	0,38	13	0,20	0,87	0,32				
20	18	0,33	0,90	0,49	18	0,27	0,90	0,42				
25	21	0,39	0,84	0,53	21	0,32	0,84	0,46				
30	24	0,44	0,80	0,57	24	0,36	0,80	0,50				

The analysis of the accuracy of the recommendations made to the four users is shown in Figure 12 to Figure 15 (note that Figure 15 visualises the results for the last user, which are detailed in Table 19,

Table 20 and Table 21). These four figures show the value of F1, which is the harmonic mean of the precision and the recall (which are obtained by comparing the recommendations of the systems with the ideal list of recommendations for each user). Each of the figures has 3 graphics, which show the results obtained considering 1, 2 or 3 clusters in the recommendation process (the more clusters are considered, the bigger is the number of recommended cities, as shown in Table 19,

Table 20 and Table 21). The x-axis of each graphic represents the maximum relative distance to the profile allowed for a city to be recommended (from 0.1 to 0.5); thus, the bigger the distance, the larger will be the number of recommended cities (see also Table 19,

Table 20 and Table 21). Finally, each graphic has 6 lines, which correspond to the results obtained for an expected number of 5, 10, 15, 20, 25 or 30 recommendations.

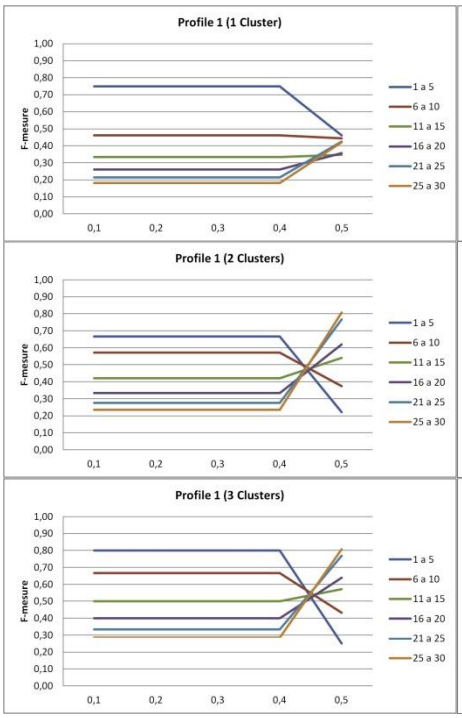


Figure 12. F1 score results of the recommendation for the profile 1

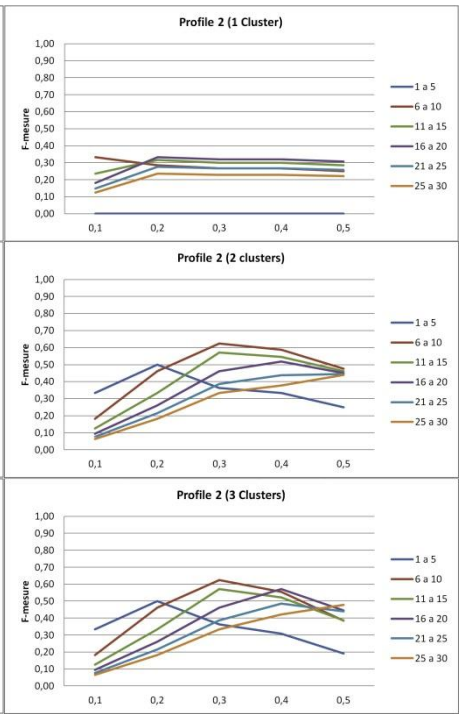


Figure 13. F1 score results of the recommendation for the profile 1

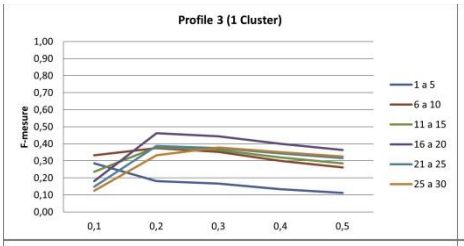


Figure 14. F1 score results of the recommendation for the profile 3

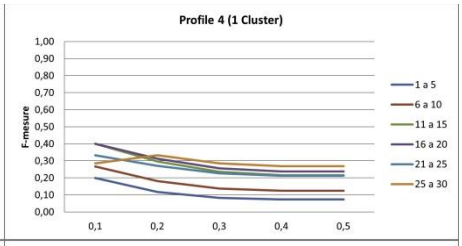
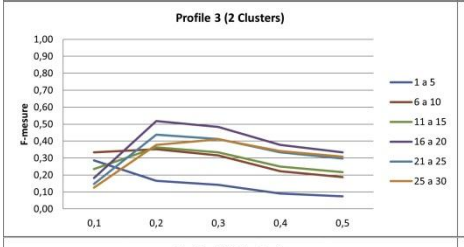
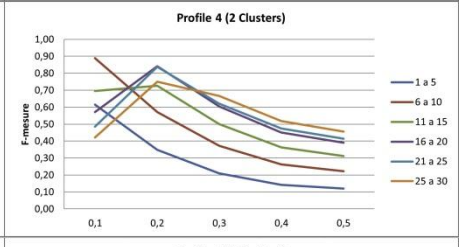
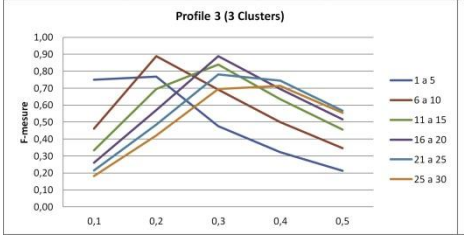
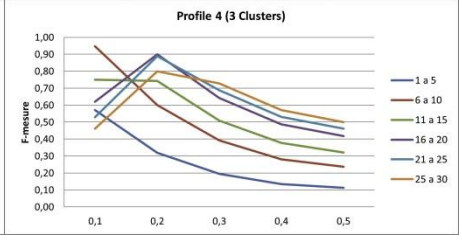


Figure 15. F1 score results of the recommendation for the profile 4









Analysing the values of *F1* in Figure 12 to Figure 15, it is possible to observe different behaviours in the four profiles that have been taken as case studies. Some conclusions that can be reached from these results are the following:

- In the case of the user that presented a very specific set of requirements (profile 1), the recommender achieves high F1 values in the most restrictive setting, when it considers a single cluster, a very close intra-cluster distance of 0.1 and a low number of expected recommendations (5). Concretely, in this case the F1 score is 0.75, and the precision was 1. This fact is due to the fact that only a few cities satisfy the requirements of the user. These results show that the proposed distance measure has been able to group the similar cities in the same cluster, because taking into account only 1 cluster and the shortest distance of 0.1 the recommender has identified the cities that satisfy the required conditions. The results obtained for profile 1 also show that the F1 score is stable until a distance of 0.4 and decreases for higher values. In a similar way, the F1 score decreases when the number of expected recommendations is increased, because the

system is forced to recommend more cities, including those that do not fit with the preferences of the user, reducing the precision and the recall of the recommendation.

- The results for the two users that had an intermediate set of requirements (profiles 2 and 3) are similar, but in the case of profile 3 we obtain better results for the same distance values and number of expected recommendations. The best F1 scores are obtained when the recommender uses the 3 clusters closer to the profile, maximum intra-cluster relative distances between 0.2 and 0.3 and up to 10 recommendations (concretely, a F1 score of 0.89 with a precision of 1). The explanation is twofold. On the one hand, as there are more cities that meet the requirements of the user, the precision is increased when more recommendations are made. On the other hand, as the required values are quite common and they appear in many cities (e.g. swimming or cycling), the system needs to consider several clusters and to increment the intra-cluster relative distance to find all the appropriate results. However, despite the use of more clusters and the allowance of a bigger distance, the recommender does not reduce its accuracy, which shows that the proposed distance measure discriminates correctly the cities in function of their most representative attributes.

- In the case of the user with more general requirements (profile 4), the higher F1 scores are obtained with a distance 0.1 and 10-15 expected recommendations (concretely, 0.89-0.70 for 2 clusters and 0.95-0.75 for 3 clusters, as shown in figure 15 ,

- Table 20 and Table 21). There are also very good results (F1 higher than 0.70, highlighted in tables Table 19 to Table 21) with a maximum distance of 0.2 and 15 to 30 recommendations. For instance, a F1 value of 0.90 is obtained when the recommender uses 3 clusters and 20 expected recommendations (concretely, in this case the system makes exactly 20 recommendations, and 18 of those 20 cities appear in the 20 first positions of the ideal ranked list of 150 cities for user 4). However, it can be observed in Figure 15 that the accuracy of the recommendations is degraded using greater distances. The main reason is that the preferred values (park, square, house, church) are very common and they can be found in most cities. Thus, the distances between the cities are very small in this case, and it is difficult to differentiate those that fit better with the values requested in all the attributes. With an intermediate distance of 0.2 the proposed semantic measure evaluates in a suitable way the similarity between the cities.

In general, these observations suggest that, on the one hand, in order to obtain the best recommendations with the system, the maximum relative distance to the profile allowed to recommend a city should be 0.3, since bigger distances always lead to worse results. This limit is consistent with the goal of the proposed

semantic measure of evaluating the similarity between objects described with multi-valued semantic attributes. On the other hand, the appropriate number of cities to recommend is 10 and the best results are obtained using up to 3 clusters. This fact significantly reduces the number of cities to consider initially (only 10 centroids instead of the whole set of 150 cities, 6%) and limits the number of cities to treat during the computation of the recommended cities (to the ones that belong to the 3 best clusters).

4.3 Summary

One of the basic pillars of the Semantic Web paradigm is the idea of having explicit semantic information that can be used by intelligent agents in order to solve complex problems of Information Retrieval and Question Answering and to semantically analyse and catalogue the electronic contents. This fact has motivated the creation of new data mining techniques like semantic clustering (Batet, 2011), which are able to exploit semantics of data. However, these methods assume that input contents were annotated in advance so that relevant features can be detected and interpreted.

The work shown in this chapter aimed to extract and pre-process data from different kinds of inputs (i.e., plain textual documents, Web resources or semi-structured documents like Wikipedia articles) in order to generate the required semantically tagged data for the aforementioned semantic data analysis techniques. In order to reach this goal several well-known techniques and tools have been used: natural language processing parsers have been useful to analyse texts and detect named entities, Hearst Patterns have been used to discover potential subsumer concepts of named entities, and 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. The main contribution, in comparison with related works, is the proposal of a novel way to go from the named entity level to the conceptual level by using the Web as a general learning source, so that the recall of the annotation can be improved regardless of the coverage limitations of named entities presented by the input ontology or WordNet. This enables a final annotation that minimises the number of queries performed to the Web, since the final matching is performed at a conceptual level (Cimiano et al., 2004). Moreover, statistical assessors have been tuned to better assess the suitability of the extraction and the annotation at each stage, minimising the inherent language ambiguity of Web queries. The method has been also designed in a general way so that it can be applied to different kinds of inputs and exploit semi-structured information (like Wikipedia annotations) to improve the performance. Finally, being unsupervised and domain-independent, the implemented methodology can be applied in different domains and without human supervision.

The evaluation performed for different entities and domains produced usable results that, by carefully tuning the algorithm thresholds, reach the high precisions needed for applying them to data mining algorithms. This fact has been proven with an explicit case study on the recommendation on touristic activities, in which a semantic clustering of the destinations (based on the features automatically extracted from Wikipedia by the algorithm presented in this chapter) leads to a more efficient recommendation algorithm, avoiding the cost of comparing the preferences of the user with the characteristics of all possible destinations, without losing precision and recall (Moreno, Valls, Mata, et al., 2013). The evaluation has also shown the benefits of using semi-structured inputs (Wikipedia articles), producing quality results with a lower computational cost.

The main publications related to the contributions presented in this chapter are:

- Vicient, C., Sánchez, D., Moreno, A. An automatic approach for ontology-based feature extraction from heterogeneous textual resources. Engineering Applications of Artificial Intelligence 26, pp. 1092-1106, 2013.
- Vicient, C., Sánchez, D., Moreno, A. Ontology-Based Feature Extraction. In Workshop on 4th Natural Language Processing and Ontology Engineering (NLPOE 2011) in IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT 2011) (Vol. 3, pp. 189–192). Lyon (France).
- Vicient, C., Sánchez, D., Moreno, A. A Methodology to Discover Semantic Features from Textual Resources. In Sixth International Workshop on Semantic Media Adaptation and Personalization (SMAP 2011) (pp. 39–44)
- Moreno, A., Valls, A., Martínez, S., Vicient, C., Marín, L., Mata, F. Personalised recommendations based on novel semantic similarity and clustering procedures. AI Communications, accepted for publication (in press).
- Moreno, A., Valls, A., Mata, F., Martínez, S., Marín, L., Vicient, C. A semantic similarity measure for objects described with multi-valued categorical attributes. In Frontiers in Artificial Intelligence and Applications, Vol. 256, pp. 263–272, 2012.

Chapter 5
Topics in Twitter

The previous chapters have focused on ontology-based information extraction procedures, in which information items were linked to concepts (represented by ontology classes) to provide them with an appropriate semantics. The next chapter will also focus on the use of semantic techniques to extract knowledge from a corpus of documents; concretely, it will present a new method to discover clusters of semantically related Twitter hashtags, which will correspond to different topics underlying a set of tweets. As suggested in section 2.1.4, the peculiarities of the messages in Twitter make the discovery process a challenging task, in which standard Natural Language Processing procedures and statistical measures do not present a good performance. Before presenting the semantic clustering procedure, this chapter provides a survey of the three basic kinds of techniques that have been proposed to detect the main topics of interest within a set of messages exchanged in a social network (Aiello et al., 2013): *probabilistic models*, *document-pivot approaches* and *feature-pivot methods*.

Probabilistic topic models consider topic detection as a probabilistic inference problem, in which a topic is represented by a distribution of terms. A new tweet will be associated to a particular topic if it contains several terms that have a high probability of appearance in that topic. *Document-pivot* methods group together individual documents according to their similarity and they associate a topic with each set of related documents. After calculating the topics, a tweet could be linked to a topic if it is similar to the cluster members (or to a representative of the cluster). Finally, *feature-pivot* methods group together the more relevant terms appearing in the analysed documents according to their co-occurrence patterns; thus, a topic is defined by a set of terms. The relevant terms may be nouns, Noun Phrases, named entities, hashtags, user mentions, etc. Given a new tweet to be classified under a given topic, its main features should be extracted and compared to the ones associated to each topic. The proposal made in this dissertation, to be presented in the next chapter, is based on the semantic clustering of hashtags; therefore, it fits in the last category of methods. Other less usual topic detection methods (for instance *Frequent Pattern Mining* (Aiello et al., 2013)) have also been proposed. The following sections comment the main characteristics of the three basic kinds of approaches and introduce some recent related work in this area.

5.1 Probabilistic models

Probabilistic models are a suite of algorithms whose aims are to estimate (based on historical data) the probability of an event occurring again and to discover the hidden thematic structure in large archives of documents. In this area of study, some researchers have proposed *probabilistic topic modelling algorithms* which annotate large archives of documents with thematic information. Those algorithms analyse (by means of statistical methods) the words that appear in a document in order to discover the underlying topics. The main advantage of these methods is that they do not require any prior annotation or labelling of the documents because topics are supposed to emerge directly from the analysis of the original documents.

The simplest and well-known topic model is the *latent Dirichlet allocation* (LDA) (Blei, Ng, & Jordan, 2003, 2012). LDA is a statistical model of document collections that aims to capture the intuition that documents exhibit multiple topics (understanding a topic as a distribution over a fixed vocabulary). Usually, in a collection of related documents, they share the same set of general topics but each document by itself exhibits those topics in different proportions. The *observed variables* are the words of the documents, the *hidden variables* are the topic structure and the generative process is the process that defines *a joint probability distribution* over both the observed and hidden random variables. The data analysis performed over such joint distribution is known as *the posterior distribution* (or just *the posterior*) and its goal is to compute the conditional distribution of *the hidden variables* given the values of *the observed variables*. The computational problem of inferring the hidden topic structure from the documents is the problem of computing *the posterior distribution*, the conditional distribution of the topics given the documents. Unfortunately, the posterior may not be directly computed because the number of possible topic structures is exponentially large.

Despite their limitations, probabilistic generative model approaches are very useful in statistical Natural Language Processing, and that is the reason why some researchers have started to apply this kind of models in the analysis of messages in micro-blogging services such as Twitter. For example, in (Celikyilmaz, Hakkani-Tur, & Feng, 2010), the authors proposed a system able to classify tweets into two categories, polar and non-polar, being polar those tweets that express certain positive or negative sentiment towards entities or events. Sentiment classification models based on Twitter data such as this one could provide relevant real-time feedback for marketing and financial purposes. Another example is shown in (Panasyuk, Yu, & Mehrotra, 2014), in which the authors also use LDA to discover the main topics associated to the tweets sent during a certain period of time by members of the American House of Representatives, and they apply sentiment analysis methods to evaluate the positive or negative view of the Democrat and Republican parties towards those issues.

As probabilistic models of language (such as topic models) are typically driven by long-term dependencies between words, they use the LDA model to extract *semantic concepts*, understood as probability distributions over words that tend to co-occur in text. However, it may be intuitively realised that, due to the particular characteristics of tweets, co-occurrence-based models will not provide as good results in Twitter as in the study of standard long documents (this fact will also be commented in the case study presented in the next chapter, in which the extremely reduced co-occurrence of hashtags is shown). In (Rajani, McArdle, & Baldridge, 2014) they propose a variant of LDA, called the *Author-Recipient-Topic* model, in which the probabilistic distributions of words are conditioned to the document's authors and recipients. This model is shown to present better results than LDA when the number of topics is large (over 300). Wang et al. (Y. Wang et al., 2014) analyse a hashtag graph, based on co-ocurrences, to uncover the probabilistic distribution of words for each topic and the probabilistic distribution of topics for each hashtag.

Other works that use probabilistic models to analyse Twitter messages are TWITOBI (Kim & Shim, 2011) and its extension TWILITE (Kim & Shim, 2014). Both works propose a recommendation system for Twitter, using probabilistic modelling based on LDA, which recommends the top-K users to follow and the top-K tweets to read for a user. The model can capture the realistic process of posting tweet messages by generalising a LDA model as well as the process of connecting to friends by utilizing matrix factorization. In TWITOBI, the model estimates the probability that a user *u* generates a word *w* in his tweets, whereas TWILITE is an algorithm that estimates the topic preference distributions of users to generate tweet messages as well as the latent factor vectors of users to establish friendship relations. Ma et al. (Ma, Sun, Yuan, & Cong, 2014) propose the use of a related mechanism, *Probabilistic Latent Semantic Analysis*, to discover the probabilistic distribution of words and hashtags for each topic.

Other works such as the ones presented in (Quercia, Askham, & Crowcroft, 2012; Ramage, Dumais, & Liebling, 2010) use models based on LDA to characterise micro-blogs with topic models. (Ramage et al., 2010) present a supervised learning model called *Labelled LDA* that maps the content of the Twitter feed into dimensions. These dimensions, according to the authors, correspond roughly to four main categories: those about events, ideas, things or people (substance), those related to some socially communicative end (social), those related to personal updates (status) and those indicative of broader trends of language use (style). The authors claim that the advantage of using a model based on LDA is that those models have been widely applied to problems in text modelling and they don't require manually labelled data. By contrast, the proposed Labelled LDA incorporates supervision where available. Thus, they destill collections of tweets into distributions of words that tend to co-occur in similar documents (the sets of related words are referred to as "*topics*"). Eventually, the posts of individual users can be mapped to one of the four pre-defined categories, giving a mechanism to characterise users by the topics they most commonly use and allowing a personalised feed re-ranking and user-customised suggestions. Labelled LDA is also used in SCAT (*System for Concept*

Annotation of Tweets) (Sachidanandan, 2014) to annotate tweets in a batch mode, given a set of previously annotated ones. In this work each tweet is finally associated to Wikipedia articles and WordNet entries.

Some authors have investigated not only the detection of topics in Twitter but also their temporal evolution (this aspect has not been considered in this dissertation, although it would certainly be an interesting line of future work). Yang and Rim (M.-C. Yang & Rim, 2014) devised a variant of LDA called *Trend Sensitive-LDA*, which takes time into account to make a more detailed study of how topics emerge and evolve. In this work the topic candidates are selected according to three criteria: *integrity* (the words describing a topic should appear in a manually constructed dictionary), *spatial entropy* (a meaningful topic should be described with words that appear in a small number of documents) and *temporal entropy* (the words associated to a topic should be concentrated in a short amount of time). A candidate topic is selected when it has a high integrity and low spatial and temporal entropies. LDA was also used by the visualisation system TweetViz (Stojanovski et al., 2014) to show the temporal evolution of topics in a graphical way, as shown in the following figure (Figure 16). The left side of the figure shows the temporal evolution (on the x-axis) of 10 topics (shown in the y-axis), whereas the right side gives the details on the distribution of probabilities of the 10 topics on a particular point in time.

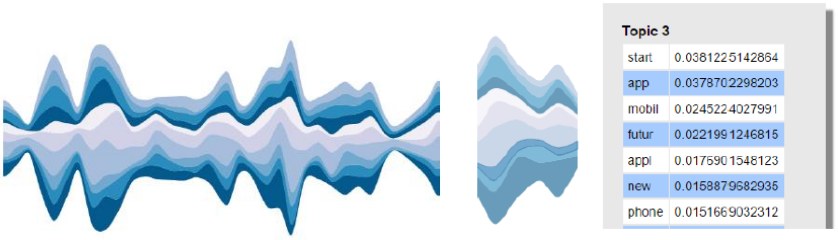


Figure 16 Visualisation of the temporal evolution of topics in TweetViz.

One of the main shortcomings of LDA-based models is their high computational cost when they have to manage a large dataset; in consequence, they must be parallelised to scale with the size of the dataset. Another limitation of LDA is that the number of topics is assumed to be known and fixed in advance, and that is certainly a very strong assumption in the case of Twitter datasets. Furthermore, the quality of the results mostly depends on the size of the training dataset: the bigger the data corpus, the higher the performance is. In general, it may be argued that probably the most important shortcoming of LDA-based models is that they are bassed on the assumption that a topic may be identified by the common co-occurrence of a particular set of words. This hypothesis, which is probably correct in large sets of big documents, fails when short, noisy, informal, ungrammatical and uncontextual messages like tweets are considered. For instance, Aiello et al. compare different topic detection methods in their detailed survey (Aiello et al., 2013), and LDA shows a very low precision in the discovery of relevant keywords (although it presents a good recall).

5.2 Document-Pivot methods

Topic detection methods based on document-pivot approaches are driven by a process of clustering of related documents. Each of the obtained clusters is supposed to represent a given topic, and the documents that belong to the class are taken to refer to that topic.

These methods require a measure of similarity between documents (and also between a document and the prototype or centroid of a cluster). When a new document *d* has to be classified in a cluster, two approaches (or small variants of them) are usually followed:

- It can be compared with all the documents that have been already classified, and it can be added to the cluster to which the most similar document belongs.
- It can be compared with the centroids/prototypes of the existing clusters, and it may be added to the cluster represented by the most similar centroid.

In both cases, the similarity between *d* and the previous documents/centroids must be above a given threshold; otherwise, a new cluster should be created for this new document. There are works that even try to combine both mechanisms (e.g. the breaking news detection algorithm proposed in (Phuvipadawat & Murata, 2010)).

The difference among the methods in this family resides in how each document/centroid is represented and how the similarity between them is calculated. A very usual approach (Fang, Zhang, Ye, & Li, 2014; Godfrey, Johns, Meyer, Race, & Sadek, 2014; Ifrim, Shi, & Brigadir, 2014; W.-J. Lee, Oh, Lim, & Choi, 2014; Phuvipadawat & Murata, 2010; Tsur, Littman, & Rappoport, 2012, 2013; Veltri, 2012) is to represent each tweet as a bag of words/terms/n-grams, where each one is weighted using the standard *tf-idf* value, and the cosine distance between two tweets is employed to measure the co-occurrence of their terms. Some authors also employ the *tf-idf-based* representation but they consider different ways of classifying tweets, using classical clustering methods like *k-NearestNeighbours*, *Support Vector Machines* (H. Rosa, Batista, & Carvalho, 2014) or *Matrix Factorization* (Godfrey et al., 2014). In some works they consider the use of different distance functions; for instance, Yi (Yi, 2013) uses the standard Jaccard measure to compare tweets with the subject headings of the Library of Congress to classify them. Some authors boost the relevance of special components of tweets, such as named entities or hashtags, so that they have a higher weight in the computation of the similarity between documents (Ifrim et al., 2014; Petkos, Papadopoulos, & Kompatsiaris, 2014). In a topic modeling method proposed by Twitter itself (S.-H. Yang, Kolcz, Schlaikjer, & Gupta, 2014) each tweet was represented with the hashed values associated to each group of four

consecutive bytes (i.e., 4-grams) and a regression model was considered to classify them to a topic of a predefined ontology of 300 topics.

All these methods associate a topic to each cluster of tweets. Figure 17 shows an example of the main topics discovered after analysing 30,000 tweets related to the 2014 football World Cup (Godfrey et al., 2014).

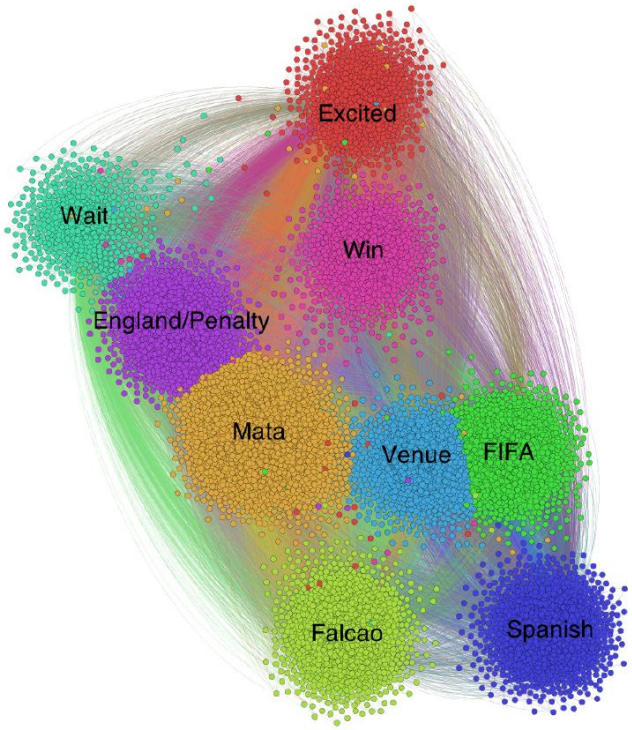


Figure 17 Topics related to the 2014 World Cup (Godfrey et al., 2014)

These methods basically rely on the co-occurrence of terms, so they usually work much better with long documents (e.g., Web pages, news articles, scientific papers, Wikipedia entries) that with tweets. Some proposals try to solve this problem by adding a pre-processing stage in which groups of temporally or topically related tweets are merged into a super-document which is used in the clustering process. For instance, in the *TwitterCrowds* system (Archambault, Greene, & Cunningham, 2013) all the tweets of a given user during a certain period of time are concatenated. (W.-J. Lee et al., 2014) use the nouns contained in all the tweets of a user to build its profile, which is employed by a recommender system to detect the news that fit better with the user's interests. Tsur et al. (Tsur et al., 2012, 2013) and (Y. Wang et al., 2014) create *non-sparse virtual documents* by concatenating all the tweets that share a common hashtag.

Panasyuk et al. (Panasyuk et al., 2014) aggregate the tweets that have the same hashtag or contain a mention to the same user. Ferrara et al. (Ferrara et al., 2013) define *protomemes* as the sets of tweets that share a hashtag, a user mention, a URL or a term in their content. They consider different notions of similarity between protomemes (e.g. common users or common tweets), which are used to make a hierarchical clustering of these sets. Aiello et al. (Aiello et al., 2013) work with *super-documents*, built by gluing together tweets that are contiguous in time or near-duplicate tweets posted in the same time slot. In the event detection method described in (Kunneman & van den Bosch, 2014) the tweets that share a unigram (in their case a Dutch word) are also concatenated.

Other works utilise a variant of the incremental clustering approach, termed "*leader-follower*" clustering, which takes into account both the textual and the temporal proximity between an incoming tweet and each cluster. In order to have a scalable approach those methods usually do not examine all existing clusters, but only those ones which share at least one textual feature with the incoming tweet (Petrović, Osborne, & Lavrenko, 2010; Sankaranarayanan, Samet, Teitler, Lieberman, & Sperling, 2009).

In any case, in most of the approaches to topic detection in Twitter described in the literature the definition of similarity between tweets is purely syntactic (unlike the one that will be presented in the next chapter of this dissertation, which has a strong semantic component). For instance, Russell et al. (Russell, Flora, Strohmaier, Poschko, & Rubens, 2011) analyse sets of tweets related to the Energy domain, and they define a notion of "semantic similarity" between tweets which merely considers the co-occurrence of their terms. Bhulai et al. (Bhulai et al., 2012) also consider the co-occurrence of words within tweets to cluster them for visualization purposes. Teufl and Kraxberger (Teufl & Kraxberger, 2011) represent a tweet with a set of terms (nouns, adjectives, verbs and hashtags) and they use the co-occurrence between them to define a weighted graph that can be analysed to obtain the "semantic pattern" associated to each tweet. Veltri (Veltri, 2012) also uses the co-occurrence between words to classify tweets related to the Nanotechnology domain. Dela Rosa et al. (K. Dela Rosa, Shah, Lin, Gershman, & Frederking, 2011) consider the words that appear in the tweets to classify them into a predefined fixed set of general categories. The lack of a semantic treatment of the content of the tweets, including their hashtags, and the reliance on the size of the analysed dataset are the main shortcomings of all these approaches. Also, as stated before, some of the proposed methods present some scalability problems and documents may be clustered incorrectly because the similarity between pairs of documents may be caused by noisy features (G. P. C. Fung, Yu, Yu, & Lu, 2005). Finally, these document-pivot approaches rely on a similarity threshold that should be carefully set. If the threshold is too low, a cluster could merge different topics; on the other side, if it is too high, a topic could be fragmented in different clusters (merging and fragmentation problems will be discussed in more depth in section 5.4). Some researchers have proposed different methods to improve the clustering process and try to avoid these problems (Becker, Naaman, & Gravano, 2011; Sankaranarayanan et al., 2009).

5.3 Feature-Pivot methods

Feature-Pivot methods differ from Document-Pivot approaches in the fact that instead of clustering documents to represent topics they cluster together *terms* (usually based on their co-occurrence patterns) and each group of terms becomes a topic (e.g. the graph-based feature-pivot method defined in (Cataldi, Di Caro, & Schifanella, 2010)). In general, feature-pivot methods involve two steps. The first one is the selection of the terms that will be eventually clustered. Different criteria may be applied in this selection step. In the second one, clustering procedures are applied on the set of selected terms, using some kind of similarity measure between pairs of terms.

In the case of Twitter, these methods focus their attention on some relevant components (or "*features*") of the tweets (e.g. hashtags, named entities, URLs, mentions, terms, etc.) and make a clustering of these elements and not of the tweets themselves. Each topic is finally represented by a set of closely inter-related terms. In particular, several authors have focused on the analysis of *hashtags* (as proposed in this paper), using the co-occurrences between them as a similarity metric, but they seem to take a purely syntactic point of view. Some recent examples of this kind of approaches are the following: Tsur et al. (Tsur et al., 2012) construct, for each hashtag, a vector with the words contained in the tweets in which the hashtag appears. After that, k-means is used to cluster these vectors in order to obtain sets of related hashtags. Wang et al. (X. Wang, Wei, Liu, Zhou, & Zhang, 2011) define a sentiment analysis method based on different mechanisms of sentiment propagation in graphs that basically take into account the co-occurrence of hashtags. Ozdikis et al. (Özdikiş, Şenkul, & Oguztüzün, 2012) cluster hashtags by considering the co-occurrence between the hashtags and the words that appear in the tweets. Pöschko (Pöschko, 2011) also considers the co-occurrence between hashtags to group together related tweets for visualization purposes. A similar approach is followed by Alfayez and Joy (Alfayez & Joy, 2013), that associate clusters of co-ocurring hashtags with terms in the OPD directory. Cotelo et al. (Cotelo, Cruz, & Troyano, 2014) create a graph whose nodes represent users and hashtags, and apply the PageRank algorithm to detect the more relevant nodes, which indicate the main topics of the set of tweets. In (Cataldi et al., 2010) the authors propose a methodology that selects and clusters terms using their "energy", which takes into account both their frequency and the importance of the users that have posted documents including the term. Only the terms with the highest "energy" are clustered using a graph-based algorithm.Dreer et al. (Dreer, Saller, & Elsässer, 2014) take the hashtags present in German news and cluster them, using as distance measure a simple frequency distribution over their co-ocurrence.

The combination of different term selection criteria, inter-term similarity measures and clustering procedures has produced a large variety of feature-pivot approaches in the literature. As stated in (Aiello et al., 2013), most of the proposed static topic models are based on LDA. Although some LDA extensions (Blei &

Lafferty, 2006) have been proposed for the analysis of dynamic data, there are also approaches that aim to capture topics through the detection of keyword *burstiness*[11] (Shamma, Kennedy, & Churchill, 2011). These methods have been widely used in real-time environments in which the discussion topics evolve quickly in time (for example breaking news) and they reach a fast peak of attention from social media users as soon as they are publicly announced. Following the main steps of feature-pivot approaches, the idea is to first identify the *bursty* terms in a text stream (i.e a sequence of chronologically ordered documents) and then cluster them together (e.g. (Stilo & Velardi, 2014; X. Zhang et al., 2014)). This area of study is sometimes called *event detection and tracking*, rather than *topic detection* (Lavanya & Kavipriya, 2014). Notice that these two problems differ in the fact that the first one attempts to discover hot *bursty* events (i.e a minimal set of terms that occur together very frequently in a certain temporal window with a strong support of the documents in the text stream) whereas the second one is a more general approach that detects permanent topics (instead of specific temporal events) in a fixed dataset.

The works in which the selection of features and their clustering are based on co-occurrences usually implement a graph-based feature-pivot approach where each node represents a term (with a frequency above a certain threshold) and each edge is weighted by some measure of inter-term correlations. One of the usual problems of these approaches is that, when there are many related topics in the analysed corpus, they may produce low quality results because they take into account only pairwise co-occurrence patterns and they may not be able to identify finer topics. The work presented in (Petkos, Papadopoulos, Aiello, Skraba, & Kompatsiaris, 2014) precisely studied the effect of the "degree" of examined co-occurrence patterns on the term clustering procedure and concluded that a feature-pivot topic detection method should examine co-occurrence patterns of degree larger than 2 when dealing with corpora containing closely inter-related topics.

5.4 Summary

Each one of the three approaches has advantages and disadvantages with respect to the others and it is not easy to conclude which one of them gives the best results. Some studies, like the one presented in (G. P. C. Fung et al., 2005) conclude that document-pivot methods may cluster documents incorrectly because the similarity between pairs of documents may easily be dominated by noisy or irrelevant features. Probabilistic approaches have been reported to produce good

[11] If a term is used once in a document, then it is likely to be used again. This phenomenon is called *burstiness*, and it implies that the second and later appearances of a word are less significant than the first appearance. http://cseweb.ucsd.edu/~elkan/perplexity.html Last access: November 10th, 2014.

results (especially when recall is considered, (Aiello et al., 2013)), but they are typically quite computationally expensive and they require a large dataset.

The main common shortcoming of these methods is that they lack a *semantic* treatment of the terms; thus, they are unable to understand the meaning of an acronym, they may fail to identify different spellings or abbreviations of the same term, they may easily not group together synonym terms (because it is very unlikely that they will co-occur in a very short statement like a tweet), they will not be able to differentiate the diverse meanings of a polysemic term, etc. All these issues may lead to two common problems: fragmentation and merging.

Fragmentation occurs when the same topic is represented by different clusters (i.e. the topic is in some way divided into sub-topics). One of the main causes of fragmentation is a wrong configuration of the similarity thresholds used by clustering procedures. If the required similarity to put a tweet (or a feature, e.g. a hashtag) within a cluster is too high, related terms could be split in different clusters. Despite the fact that fragmentation is a common problem of all the approaches, it is likely to be more pronounced in document-pivot methods due to the fact that the same concept may be expressed in several ways in different documents (e.g. synonym terms may be used in two tweets and the clustering procedure would not be able to detect this fact). A similar phenomenon can happen in feature-topic approaches (e.g. lexically different hashtags may be easily referring to the same concept). Some authors attempt to deal with fragmentation by applying a second step in the clustering procedure to find topics fragmented in several clusters (Becker et al., 2011; Sankaranarayanan et al., 2009).

Merging is the opposite problem of fragmentation, in the sense that it occurs when many different topics are represented by a single cluster. It may happen that the resulting cluster represents either a set of related lower-level topics (that could have been represented more precisely in a set of smaller clusters) or a mixture of topics unrelated to each other (for example there could be a cluster contaning terms related to biological viruses and also to Computer Science viruses). The first case may be acceptable in a given application, depending on the required granularity of the topics, whereas the second case produces clearly inconsistent results that should be avoided. Merging may occur when polysemic terms are considered, or when the minimum similarity required to add a tweet (or a feature) to a cluster is too low. From a usability point of view, fragmentation is bad because topics are redundant, whereas merging is undesirable in many cases since the results are usually incomprehensible topics.

The new approach to topic detection presented in the following chapter of this dissertation tries to address these issues by making a semantic analysis of the hashtags, and not a syntactic one based on their lexical appearance. One of the basic ideas of this work, that differentiates it from the previous approaches based on syntactic co-occurences, is the establishment of a mapping between the hashtags and WordNet concepts, which is later leveraged to define a semantic similarity between them. Some related works have been made by Meij (Meij, Weerkamp, & de Rijke, 2012), that propose to associate Wikipedia concepts to

tweets by studying all their n-grams, and by Cantador (Cantador, Konstas, & Jose, 2011), that associate tags to a restricted predefined set of concepts via their mapping to concepts in Yago (which includes information harvested from WordNet and Wikipedia). The proposal described in the next chapter is more general, as the aim is to uncover all the basic topics associated to a set of hashtags, without any restriction to a small predefined set of categories. Moreover, none of the previous works that classify tweets or hashtags has a selection procedure that can identify automatically which of the obtained classes are really relevant according to the expert/user preferences. Some works (Tsur et al., 2012) apply clustering techniques like k-means, in which the number of topics to obtain has to be known in advance. Other authors (Ifrim et al., 2014) make a hierarchical clustering but they simply cut the tree at a fixed predetermined level to obtain a set of clusters. This is a very important issue when dealing with hashtags because, by their nature, they have a very large proportion of noisy information that must be discarded. In this proposal, as will be shown in section 6.2.3, a dynamic analysis of the tree is performed to detect relevant clusters at different levels of generality.

Chapter 6
Unsupervised topic discovery in micro-blogging networks

Social micro-blogging networks such as Twitter provide an enormous amount of daily information, and its automated and unsupervised analysis constitutes an exciting research challenge in which Artificial Intelligence methods may be applied. The main limitation of this kind of environments is that they present a reduced context. On the one hand, this means that traditional information extraction methodologies based on statistics are not useful in these types of resources. However, on the other hand, micro-blogging networks provide a certain kind of semi-structure that can be used to improve the information extraction process.

The previous methodology (presented in Chapter 4) focused its attention on the detection of the most relevant Named Entities of a document and on its semantic annotation in order to extract the main features that describe it, giving us a notion of the main topics or characteristics of a particular entity. Unfortunately, micro-blogging documents are strings of up to 140 characters that may basically contain text and links, so it is unlikely that they contain enough Named Entities that can characterise their main topics. This chapter presents a new unsupervised and domain-independent methodology which is able to deal with the limitation of this kind of resources taking profit of the semi-structure provided by hashtags, performing a semantic clustering of them that can be regarded as a first step towards the automatic discovery of the topics associated to a set of tweets. The methodology was tested on the field of Oncology.

6.1 Introduction

Micro-blogging services such as Twitter constitute one of the most successful kinds of applications in the current Social Web. Every day more than 500 million tweets are sent, providing up to date information about any imaginable domain of knowledge (Twitter, 2014). Each tweet is a string of up to 140 characters that may basically contain text, links, user mentions and *hashtags* (strings preceded by the #

symbol with which users tag their messages). In the last years there has been a growing interest in the design and development of tools that allow users to analyse large unstructured repositories of user-tagged data in order to discover and extract meaningful knowledge from them (Aiello et al., 2013; Teufl & Kraxberger, 2011). The determination of the main topics of interest in a collection of tweets may be a useful first step to sort them and address the problems of data visualisation, semantic (not keyword-based) information retrieval, information extraction, detection of users with similar interests, hashtag recommendation, etc. (Bhulai et al., 2012; Cotelo et al., 2014; Kywe, Hoang, Lim, & Zhu, 2012). One of the main uses of hashtags is the categorisation of tweets, because (ideally) all the tweets that share the same hashtag should somehow refer to the same topic (e.g. the tweets with the hashtag #WorldCup2014 are related to facts, events, comments or opinions about the Football World Cup in Brazil in 2014). Thus, one of the working hypotheses of this research is that the automated clustering of the hashtags present in a set of tweets may lead to a straightforward discovery of its main topics. However, grouping hashtags automatically in an unsupervised way turns out to be a very complex task, even if all the tweets belong to a certain domain of discourse (e.g. Oncology, the area of the case study developed in section 6.3).

There are two main reasons that hamper the construction of groups of related hashtags. The first one is that users can freely annotate tweets without any restriction in their choice of hashtags. This means that, by nature, hashtags are unstructured and unlimited. They lack any form of explicit organization or normalization and, as a consequence, retrieval tasks and classification methods have to deal with basic problems like synonymy (different hashtags might have been used for the same concept, e.g. #illness and #disease) or polysemy (the same tag can have different meanings in different contexts, e.g. the term #operation may refer to "surgical treatment" but also to "the act of causing to function", "an action", etc.). There may also be lexically similar hashtags that do not have exactly the same meaning (#pharmaceuticals, #pharmaceutical, #pharmacy, #pharmacology, #pharma); thus, standard stemming techniques used in Natural Language Processing may lead to wrong results. Moreover, tags may also be acronyms (#HIV - human immunodeficiency virus, #AIDS - acquired immunodeficiency syndrome), named entities (#MayoClinic, #AustinCancerCenter), a combination of several words (#HighBloodPressure), an expression of a feeling (#CancerSucks), or just invented words or even pure nonsense. All these issues present a big challenge and most of the topic discovery methods described in the current literature are not able to deal with them.

The second reason, as will be shown in the next section, is that current hashtag clustering methods are mostly based on a syntactic analysis of their co-ocurrence (Carlos Vicient & Moreno, 2013). This kind of analysis presents several problems, just to name a few:

- As tweets are very short, it is uncommon to use more than one hashtag in a tweet; in fact, some studies indicate that roughly 16% of them

contain at least one hashtag (Mazzia & Juett, 2011). Therefore, the hashtag co-ocurrence matrix is usually very sparse.

- A purely syntactic analysis will always treat a polysemic hashtag in the same way, without distinguishing its different meanings.
- Synonymous hashtags will hardly co-occur and they will not be assigned to the same cluster. For example, the terms "car" and "automobile" will unlikely appear together in the same sentence (especially with a length up to 140 characters).
- The meaning of acronyms will not be taken into account.
- The components of a multi-word hashtag will not be separately considered (e.g. the relation between #Cancer and #LungCancer will not be obvious, as they will just be treated as two different strings).
- General concepts and named entities will be analysed in the same way, as mere strings of characters.

The main hypothesis of this chapter is that the incorporation of semantic information, i.e. the analysis of the actual meaning of the hashtags, may help to alleviate these issues and to make a better clustering of them, which will lead to an improved identification of the topics underlying a tweet set. The linkage between a term (e.g. a hashtag) and its meaning (a concept in a background knowledge structure, typically a domain ontology) is called *semantic annotation* according to the Semantic Web paradigm (Berners-Lee & Hendler, 2001). Having solved this task, one may apply an ontology-based semantic similarity measure to group related terms.

The new topic discovery method proposed in this paper is thus based on the semantic annotation of hashtags supported by well-known knowledge repositories like WordNet and Wikipedia. The contributions of this chapter are threefold:

- A novel procedure to link hashtags to WordNet synsets is defined.
- A new methodology to perform an automatic unsupervised semantic clustering of the set of hashtags contained on a given set of tweets is proposed.
- It is explained how to analyse the resulting hierarchy in order to identify the classes that are really significant, filtering the huge amount of noise present in a hashtag set.

The rest of the chapter is structured as follows. Section 6.2 explains the new methodology of analysis, which is composed of three basic steps: mapping hashtags to concepts, clustering hashtags according to the semantic similarity between their associated concepts, and filtering the relevant classes of hashtags. Section 6.3 presents an application of the methodology to a corpus of tweets related to Oncology, in which encouraging results have been obtained. The final section makes a general discussion of this new topic detection method.

6.2 Methodology

This section describes the new methodology that has been designed to obtain automatically the main topics of interest of a given set of tweets, given the information provided by their hashtags. The three steps of the analysis are explained in the following subsections: *semantic annotation*, *semantic clustering* and *topic selection*.

6.2.1 Semantic annotation

The aim of this stage is to fill the gap between hashtags and concepts in order to be able to compare two different hashtags at the semantic level. *Concepts* can be considered as abstract representations of classes of objects. In Computer Science it is usual to represent the information about a particular domain in an *ontology*, which, has explained in section 2.2.1, is a knowledge structure that basically contains the domain concepts (represented as classes), the taxonomic and non-taxonomic relationships between them, their attributes and particular instances of the classes (Gruber, 1995). WordNet, introduced in section 2.2.2, is a broadly used general purpose ontology (Fellbaum, 1998) in which each term (Figure 18-a) is represented by a WordNet entry (Figure 18-b) composed by a set of concepts called *synsets* (Figure 18-c), which specify the different meanings of the term

Figure 18 WordNet entry for the term *cancer*

Thus, the idea of this first step of the analysis is to link hashtags to WordNet synsets. In this way, in the posterior clustering process it will be possible to apply ontology-based semantic similarity measures to determine the alikeness of pairs of hashtags. The pseudo-code of the semantic annotation phase is shown in Algorithm 3.

Algorithm 3 Semantic annotation

```
1 getCandidates(hashtag h) {
2     candidates := getWNConcept(h)
3         if candidates == null
4             candidates := getWikipediaCandidates(h)
5     return candidates
6 }
```

The first step (*getWNConcept*) checks if the hashtag matches directly a WordNet entry. It may happen that the term has a precise meaning (e.g. a single synset associated to #lymphoma can be directly found in the WordNet entry for the term "lymphoma"). It may also be the case that this step returns a *list* of synsets associated to a polysemic hashtag (e.g. as seen in Figure 18, #cancer may refer to the malignant neoplastic disease or to the zodiac sign, among other possibilities). If the hashtag is composed of multiple words, the function *getWNConcept* applies word-breaking techniques to split them, and then it drops sequentially the leftmost terms until a matching is found (e.g. #TerminalCancer would be separated in the words Terminal and Cancer; "Terminal Cancer" would not be found in WordNet, but "Cancer" would return a positive result). In English it is usual that the words on the left are adjectives or terms that denote a specialisation of the main noun, located on the right. Therefore, this procedure finds the most general specialisation present in WordNet.

There are many different types of hashtags that do not appear in WordNet directly (most acronyms, named entities, multi-word terms, etc.). In those cases the support of Wikipedia (*getWikipediaCandidates*) is a useful alternative procedure to find an appropriate semantic annotation. Wikipedia, as introduced in section 2.1.3, is a well-known free online encyclopaedia which contains more than 30 million articles that have been written collaboratively by volunteers all around the world. In many knowledge-based tasks associated to Natural Language Processing it has been assumed in the last years that Wikipedia is one of the most comprehensive repositories of textual knowledge, due its enormous breadth and the quality of its collaborative-based contents (Fuentes-Lorenzo, Fernández, Fisteus, & Sánchez, 2013; Romero, Moreo, Castro, & Zurita, 2012; Suchanek, Kasneci, & Weikum, 2008). Its articles are not limited to the standard <concept, succint definition> structure of a dictionary, but they contain detailed explanations of all kinds of topics. Thus, Wikipedia entries have a much wider scope than WordNet concepts, permitting to find for example acronyms, named entities, different lexicalizations of the same concept, etc. Moreover, Wikipedia articles are loosely classified by means of a hierarchy of *Wikipedia categories*. Each of them

defines the essential characteristics of a certain topic, providing readers a mechanism to browse and quickly find sets of related pages.

Algorithm 4 presents the pseudo-code of the *getWikipediaCandidates* function. If there is a Wikipedia entry for the hashtag, all the associated categories are retrieved. A category is treated as a phrase, and a parser is employed to detect its nouns. If a noun has a WordNet entry, its synsets (i.e., concepts) are taken as annotation candidates. Thus, all the proposed candidates are WordNet concepts. This annotation process guided by Wikipedia may be especially useful to deal with hashtags that represent named entities (specific individuals). For example, #Monsanto does not appear in WordNet, but it can be found in Wikipedia, where it is related to 10 categories, including "Chemical companies of the United States" and "Genetic engineering and agriculture". Thus, this hashtag would be linked to the concepts associated to terms such as "Company" and "Engineering".

Algorithm 4 Extraction of candidate concepts via Wikipedia categories

```
1  getWikipediaCandidates(hashtag h){
2     wikiCandidates := null
3     if existsWikiEntry(h)
4        auxCategories := getCategoriesFromWiki(h)
5        forall cat ∈ auxCategories
6           mainNoun := getNN(cat)
7           auxCat := getWNConcept(mainNoun)
8           if auxCat != null
9              wikiCandidates ← auxCat
10    return wikiCandidates
11 }
```

It can be noticed that a hashtag will not have any associated candidate concepts only if it is not found either in WordNet or in Wikipedia, or if the main nouns of the categories found in Wikipedia do not match with any concept in WordNet. Hashtags that have not been annotated are discarded from the analysis at this point, as the system is not able to discover any possible meaning for them. In general, at this point a hashtag will be linked to a set of WordNet synsets.

6.2.2 Semantic hashtag clustering

The aim of this second stage of the methodology is to group all the similar hashtags in clusters of related terms in order to detect topics of interest. Clustering procedures need to know the similarity between each pair of hashtags. The *semantic similarity matrix* (S_n) is defined in Eq. 9, where n represents the total number of annotated hashtags and s_{ij} is the semantic similarity between the i_{th} and j_{th} hashtags, calculated with the *SemanticSimilarity* function (Algorithm 5), so that $\forall i \in [1,n] \forall j \in [1,n]\ s_{ij} = \text{SemanticSimilarity}(h_i, h_j, smeasure)$.

$$S_n = \begin{bmatrix} s_{11} & s_{12} & s_{1n} \\ \cdots & \cdots & \cdots \\ s_{n1} & s_{n2} & s_{nn} \end{bmatrix} \mid s_{ij} \in [0,1] \quad (9)$$

Algorithm 5 Semantic similarity between two hashtags

```
1  SemanticSimilarity(hashtag h1, hashtag h2, function smeasure){
2    LCh1 := getCandidates(h1)
3    LCh2 := getCandidates(h2)
4    if (LCh1 == null) || (LCh2 == null)
5       return 0.0
6    simMax := 0.0
7    forall conc1 ∈ LCh1
8       forall conc2 ∈ LCh2
9          sim := smeasure (conc1, conc2)
10         if(sim >= simMax)
11            simMax := sim
12   return simMax
13 }
```

Each hashtag *h* has an associated list of WordNet concepts (*LCh*), obtained in the first step with the function *getCandidates*. In order to establish the degree of alikeness between two hashtags the similarity between all the pairs of associated WordNet concepts (one from each tag) is calculated using an ontology-based semantic similarity measure (*smeasure)*. There are many semantic measures that can be used at this point, as described in section 2.3.4 (Choi & Kim, 2003; Leacock & Chodorow, 1998; Y. Li, Bandar, & McLean, 2003; Rada, Mili, Bicknell, & Blettner, 1989; Sánchez, Batet, Isern, & Valls, 2012b; Z. Wu & Palmer, 1994). It may be argued that calculating the similarity between all the pair of candidates in *LCh1* and *LCh2* and taking the maximum one solves, in an indirect way (especially when tags are associated to a certain domain of knowledge), the problem of disambiguating the correct sense of the tag. In fact, in (Tversky, 1977) this idea is discussed stating that, from a psychological point of view, people usually tend to establish the similarity between terms considering their similitudes instead of their differences. For instance, if we consider a set composed basically by medical hashtags, when the hashtag #cancer is compared with other hashtags, the candidate (synset) associated to the disease will have higher similarities than the one that refers to the sign of the zodiac.

After calculating the similarity matrix, the system has to divide the set of hashtags into clusters. It is not convenient to apply mechanisms such as k-means (Lloyd, 1982; MacQueen, 1967) that require a previous knowledge of the number of clusters to be obtained. Moreover, it is desirable to be able to detect topics of different levels of generality. That's why in this thesis a *hierarchical* clustering of the set of hashtags is calculated (Everitt, Landau, & Leese, 2001; Legendre & Legendre, 1998; Sørensen, 1948). The result of this procedure is a tree of nested clusters, from the most specific (the leaves of the tree) to the most general (represented by the root of the tree). In that way, in the next step it is possible to analyse the quality of the clusters at any point of the tree.

6.2.3 Topic selection

The methodology proposed in this chapter intends to obtain a manageable set of topics, containing only those ones that are associated to a representative number of hashtags. Each of the chosen topics will be linked to a subset of the hashtags, and there may be a large proportion of the initial hashtags that are not selected because they do not fit in any of the discovered topics (they are unrelated to the domain or they are simply not understandable by the system). In order to simplify the presentation of the results, the system will return a list of topics, and the sets of hashtags associated to each topic will be disjunct; thus, a hashtag will only belong to one of the final topics. The system also has the possibility to show the relevant topics as a hierarchy.

Notice that the hashtags that could not be linked to WordNet concepts in the first stage were dismissed, so in the second step only the annotated hashtags were clustered. Thus, a first selection of hashtags was already made in the annotation step. In this third step the objective is to analyse the hierarchical clustering of hashtags and determine the clusters that are relevant enough to represent a topic of interest. Two criteria are used in the selection process:

- The cluster should have a minimum number of elements. It is probably not interesting to select a topic that contains a very low number of hashtags.
- The set of hashtags associated to the cluster should be homogeneous enough, so that the set defines a coherent topic. The *inter-cluster homogeneity* is defined as the average semantic distance between all the pairs of hashtags of the cluster, which can be calculated with the same ontology-based semantic similarity measure used in the previous clustering process.

The filtering process may be applied in a bottom-up or top-down fashion. The first one aims to find the most especific classes that fulfill the selection criteria, whereas the second one focuses on the detection of the most general classes that satisfy them.

The pseudo-code of the *bottom-up filtering* function that selects the final non-overlapping clusters (i.e. topics) is given in Algorithm 6.

Algorithm 6 Selection algorithm (bottom-up approach)

```
1  FilteringBU (hierarchy HC, int minK,int maxK, float t1, int t2){
2    finalClusts := Ø
3    forall k in maxK .. minK
4      forall c in 1 .. k
5        b := inter-cluster-homogeneity(HC_kc)
6        if ((b >= t1) && (|HC_kc| >= t2)
7          && (∄e in finalClusts | e ⊆ HC_kc))
8            finalClusts ← HC_kc
9    return finalClusts
10 }
```

In this algorithm HC is the result of the hierarchical clustering (i.e., a tree) and *t1* and *t2* are the minimum inter-cluster homogeneity and the minimum number of elements required to select a cluster, respectively. The filtering function iteratively makes horizontal cuts in the tree, from the one that provides *maxK* clusters up to the one that gives *minK* clusters. HC_{kc} denotes the *c*-th cluster when the tree is divided into *k* classes. A cluster is selected if it is homogeneous and large enough, and it is not a superset of a previously selected class; thus, this function selects the most especific clusters that satisfy the two selection criteria. The following figure (Figure 19) illustrates the selection process. Let us assume that *t1*=0.6, *t2*=3, *maxK*=9 and *minK*=2. The system makes a bottom-up analysis, starting with the horizontal cuts that divide the set in 9 and 8 clusters. None of the clusters has 3 elements, so the filtering process moves to the next level (7 clusters). At this point there is a cluster with 3 hashtags (h3, h4 and h5). Assuming its homogeneity is 0.5, it would not be selected. When the selection process divides the set in 4 clusters, it finds a cluster with 5 elements (h3 to h7, framed in green in the figure). If this set had an homogeneity 0.6 it would be the first one selected by the algorithm. The cluster h1-h7, which appears in the next cut of the tree, would not be considered because it contains a group of hashtags that were already selected (h3-h7). Finally, when the tree is cut in 2 clusters a new cluster with 3 elements is found (h8-h10). This cluster will also be selected because its homogeneity exceeds 0.6. Note that the hashtags h1 and h2 would not appear in any of the filtered clusters.

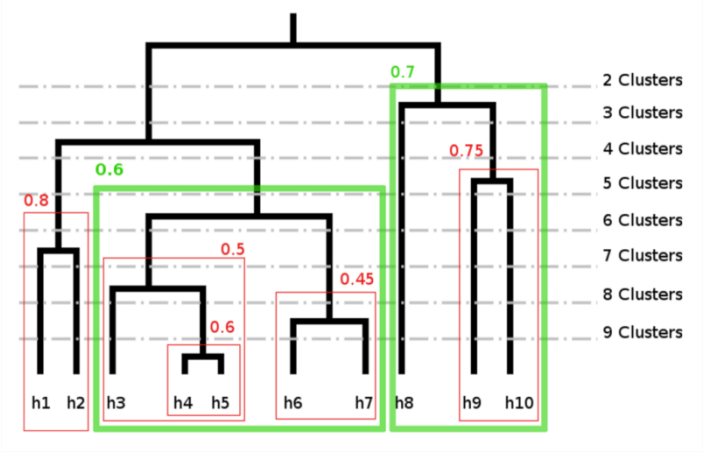


Figure 19 Bottom-up filtering process

Algorithm 7 Selection algorithm (top-down approach)

```
1  FilteringTD (hierarchy HC, int minK,int maxK, float t1, int t2){
2    finalClusts := Ø
3    forall k in minK .. maxK
4      forall c in 1 .. k
5        b := inter-cluster-homogeneity(HC_kc)
6        if ((b >= t1) && (|HC_kc| >= t2)
7          && (∄e in finalClusts | HC_kc ⊆ e))
8            finalClusts ← HC_kc
9    return finalClusts
10 }
```

In this algorithm, the filtering function iteratively makes horizontal cuts in the tree in the inverse direction (i.e. from the one that provides *minK* clusters down to the one that gives *maxK* clusters). A cluster is selected if it is homogeneous and large enough, and it is not a subset of a previously selected class; thus, the main difference with the previous approach is that this function selects the most general clusters that satisfy the two selection criteria instead of the most specific ones. The following figure (Figure 20) illustrates the selection process. Again, let us assume that *t1*=0.6, *t2*=3, *minK*=2 and *maxK*=9. The system makes a top-down analysis, starting with the horizontal cut that divides the set in 2 clusters (h1-h7 and h8-h10). Both clusters have more than 3 hashtags and they have a homogeneity greater or equal than 0.6, so both of them would be selected. The rest of the horizontal cuts from 3 to 9 clusters are all of them subsets of either h1-h7 or h8-h10. Therefore, even if the cut produces clusters with more than 3 elements and with a homogeneity over 0.6, they will be dismissed. Note that the hashtags h1 and h2 would not appear in any of the filtered clusters of the previous approach but in this second approach they belong to the first cluster. The main consequence of this selection procedure is that the clusters obtained by a top-down analysis will produce results with a higher recall than those obtained by the bottom-up analysis, whereas the last ones will present a higher precision

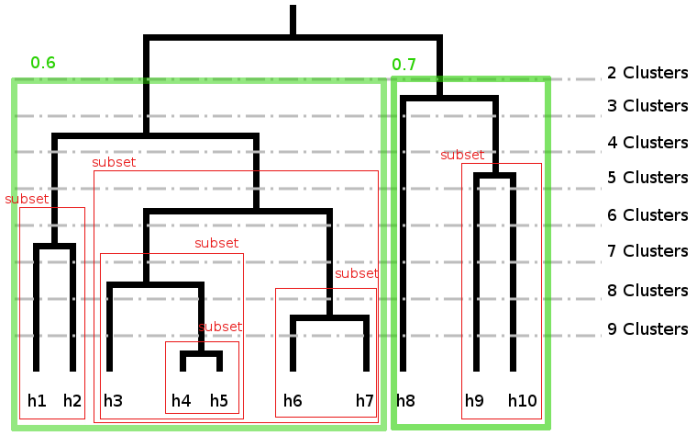


Figure 20 Top-down filtering process

Both selection procedures identify a collection of non-overlapping clusters composed by a set of highly inter-related hashtags; thus, each of these sets constitutes a *topic*. In order to find automatically a representative name for each topic, the *semantic centroid* of its associated set of hashtags is calculated on WordNet with the method described in (Martínez, Valls, & Sánchez, 2012). In a nutshell, the semantic centroid of a set of concepts of an ontology is the ontology concept that minimises the average semantic similarity distance with respect to all the members of the set. Notice that the semantic centroid may be a concept that does not actually belong to the set.

In summary, at the end of the process the algorithm returns a list of topics, where each topic is associated with a set of hashtags and with a name (a WordNet concept that represents the semantic centroid of the set). With the two selection thresholds it is possible to establish a minimum number of hashtags and a minimum degree of homogeneity for each topic.

A third possibility is to apply the selection procedure to all the classes of the dendrogram, keeping all the ones that satisfy the size and homogeneity criteria (in this case, as all the dendrogram is analysed, it would not matter if the system follows a top-down or a bottom-up approach). In that way the system would obtain a hierarchy of clusters, instead of a list. This hierarchy could contain both specific clusters (located at the bottom of the hierarchy, with a small size, high precision and low recall) and general clusters (located at the top of the hierarchy, with a larger number of hashtags, lower precision but higher recall); thus, this tree would provide to the user a more comprehensive view of potentially interesting classes at different levels of generality.

The following figure (Figure 21) shows an example of this procedure. Let us assume again that the minimum selection criteria are set in 3 members and a 0.6 homogeneity. After analysing all the horizontal cuts of the dendrogram, 5 clusters that satisfy these criteria are found. They contain the following sets of hashtags, from top to bottom:

- h1-h8 (8 hashtags, homogeneity 0.71)
- h9-h12 (4 hashtags, homogeneity 0.7)
- h3-h8 (6 hashtags, homogeneity 0.6). This cluster was contained in h1-h8 (the hashtags h1 and h2 have been dropped from the more general cluster).
- h10-h12 (3 hashtags, homogeneity 0.75). This cluster was contained in h9-h12 (hashtag h9 has been eliminated from this cluster).
- h3-h5 (3 hashtags, homogeneity 0.6) and h6-h8 (3 hashtags, homogeneity 0.61). These two clusters are a direct partition of h3-h8.

In this example a bottom-up analysis would have given as a result a list of 3 clusters (h3-h5, h6-h8, h10-h12), all of them with only 3 elements (3 of the 12 hashtags would not appear in any of these clusters). The top-down analysis would have provided the clusters h1-h8 and h9-h12, which, in this case, give a partition

of the full set of hashtags. The full hierarchical analysis not only shows all these clusters and their inclusion relationships, but also intermediate clusters that also satisfy the selection criteria (for example, the cluster h3-h8).

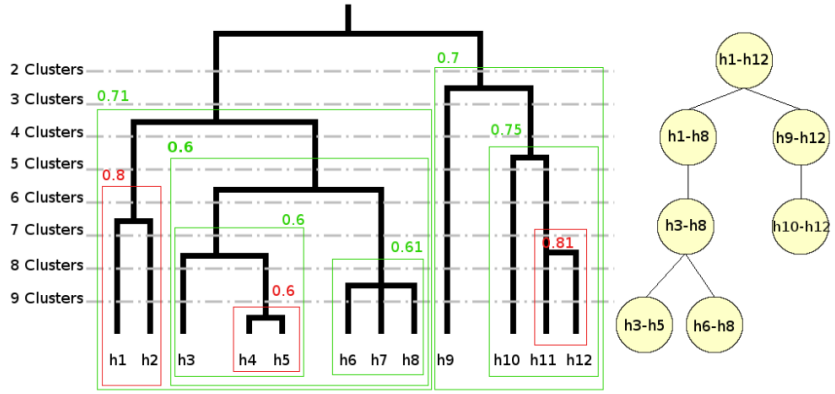


Figure 21 Example of filtered hierarchy using a bottom-up filtering analysis

6.3 Case of study

This section describes the application of the proposed methodology to a case study in which a set of tweets related to the medical field of Oncology is analysed. First, the characteristics of the data set are commented. After that, the three steps of the analytic procedure (semantic annotation of hashtags, semantic clustering and topic selection) are applied, and the obtained results are quantitavely evaluated.

6.3.1 The dataset

The dataset was manually extracted from the Symplur[12] website. It is composed of 4997 different English hashtagged tweets related to Oncology sent from October 31st 2012 to January 11th 2013. These tweets contain 1086 different hashtags.

[12] http://www.symplur.com/ Last access: November 10th, 2014

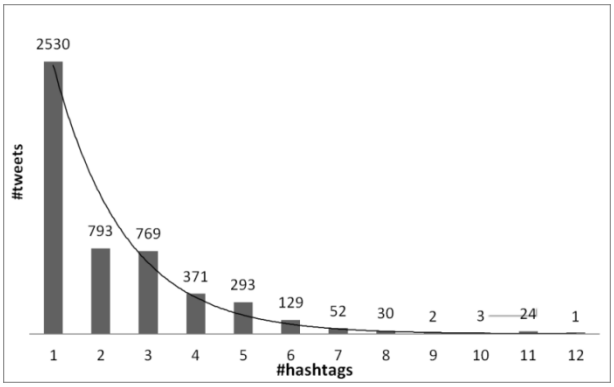


Figure 22 Distribution of hashtags per tweet

Figure 22 shows the distribution of the number of hashtags per tweet, where the y-axis represents the total number of tweets that contain exactly the number of hashtags given in the x-axis. Notice that there are 2530 tweets (50.63%) tagged only with one hashtag and just 1 tweet containing 12 distinct hashtags. Thus, around half of the tweets do not contain any co-occurrence between hashtags, and they would not contribute any information to standard clustering procedures based on them, hampering their performance. To support this idea, a preliminary study of the influence of the hashtag co-occurrences was performed by clustering all the hashtags taking in consideration only their co-occurrences (the clustering process is purely syntactic and the more frequently two hashtags appear, the more related they are supposed to be). The clustering process requires the use of a symmetric similarity matrix between hashtags. In order to generate this matrix two steps are followed. First, a hashtag co-occurrence matrix (as shown in Eq. 10) is constructed, where n represents the total number of hashtags, c_{ij} is the number of co-occurrences between hashtag i_{th} and hashtag j_{th} within the dataset, and c_{ii} contains the number of times that the hashtag i_{th} appears in the whole dataset. Notice that c_{ij} has the same value that c_{ji}, fact that implies that the order in which the hashtags appear in the set of tweets is not relevant.

$$\mathrm{C}_n = \begin{bmatrix} c_{11} & c_{12} & c_{1n} \\ \cdots & \cdots & \cdots \\ c_{n1} & c_{n2} & c_{nn} \end{bmatrix} \mid c_{ij} \in \mathbb{N} \tag{10}$$

Afterwards, the matrix has to be normalised so that all its values are between 0 and 1, since the clustering method used in this study needs either a similarity or a dissimilarity matrix as input data. To do so, each element of the matrix is normalised $(\forall i \in [1,n] \forall j \in [1,n]\, c_{ij} = NormalizedM(i,j))$ where $NormalizedM(i,j)$ is the function shown in Eq.11. This function assigns the value 1 to each element of the diagonal of the matrix (a hashtag is equal to itself). The co-occurrence between two different hashtags is normalised by dividing it by the minimum number of their individual appearances. The rationale of this approach is that two hashtags will be considered highly similar if, in most of the tweets in which one of them appears, the other hashtag also appears. For instance,

if hashtag A appears in 100 tweets, hashtag B appears in 500 tweets, and they appear together in 80 tweets, their similarity will be very high (80/100=0.8), since B appears in 80% of the tweets containing A. Other more restrictive ways of normalising the matrix values, considering for instance the maximum or the average of the individual appearances, could have also been considered.

$$\text{NormalisedM}(i,j) = \begin{cases} 1, & i = j \\ \dfrac{c_{ij}}{\text{MIN}(c_{ii}, c_{jj})}, & \text{otherwise} \end{cases} \quad (11)$$

In Figure 22 it was shown that more than half of the analysed tweets only contain 1 hashtag, and only around 18% of them have 4 or more hashtags. Therefore, the number of co-occurrences between hashtags is relatively small; moreover, the pairs of co-occurrent hashtags are likely to appear in different tweets. Therefore, there are many cells in the co-occurrence matrix with a 0 value (near to 86%), for all those pairs of hashtags that do not appear together in any of the input tweets (i.e. only about 14% of pairs of hashtags co-occur). With this co-occurrence based similarity measure, these pairs of hashtags are considered to be totally dissimilar. After appling a hierarchical clustering using the co-occurrence matrix built in the previous step, the obtained results were unmanageable since there is basically a very big cluster accompanied by a large number of very small and irrelevant clusters as seen in Figure 23.

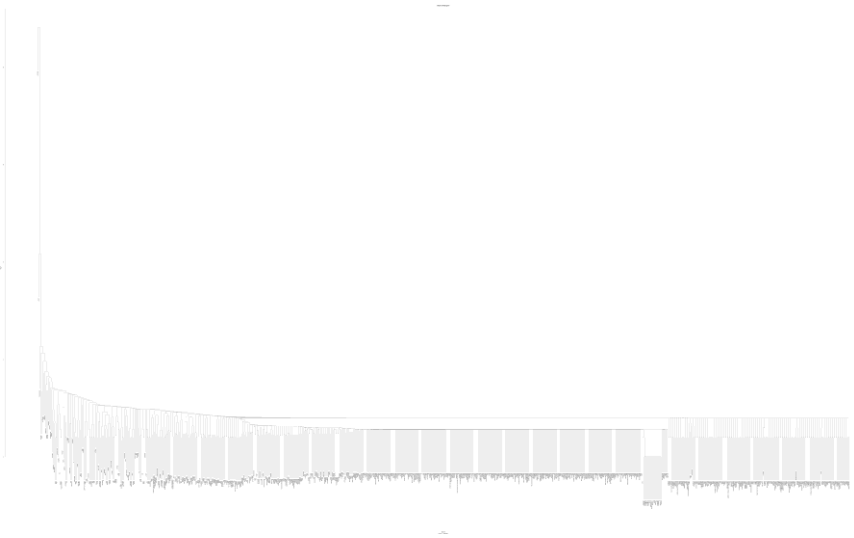


Figure 23 Co-occurrence based hierarchical clustering

6.3.2 Analysis of the set of tweets

The first stage of the analysis is the semantic annotation of the 1086 hashtags, following the procedure explained in section 6.2.1. The results were the following:

- 409 hashtags (or substrings of them, in the case of hashtags formed by a concatenation of words) were directly found in WordNet (37.66%).
- 521 hashtags were not found in WordNet, but they could be annotated with the support of Wikipedia (47.97%). Thus, a total of 930 hashtags (85.63%) were linked with at least one WordNet synset.
- The 156 remaining hashtags (14.37%) could not be annotated, and they were dismissed from the rest of the analysis.

Table 22 shows some examples of each one of these situations. The first row contains hashtags that were directly found in WordNet. Notice that there are terms represented with only one concept (e.g. hepatitis – 1 concept/synset) whereas others have more than one (e.g. cancer – 5 concepts/synsets). The second row shows some hashtags that could be annotated with the support of Wikipedia (including acronyms, Named Entities and highly specialised concepts). The last row contains some of the hashtags that could not be annotated. There are different reasons that can cause the failure of the annotation. The most common ones are the following:

- The hashtag may simply not be found either in WordNet or Wikipedia (e.g. #mucmed).
- The word-breaker may be unable to separate correctly the components of a complex term (e.g. #biobusiness).
- In many cases the hashtag is found in Wikipedia in a "disambiguation page", because it is associated to different Wikipedia articles. This situation is quite common with short acronyms (e.g. #AML has 14 different meanings, #ESMO has 2 possible meanings).

Table 22 Examples of semantic annotation

Directly found in WordNet	#cancer (5 synsets), #antibiotics (1 synset), #colon (5 synsets), #blood (6 synsets), #hepatitis (1 synset)
Found with the support of Wikipedia	#abraxane (inhibitors, breast, cancer) #tomosynthesis (breast, cancer, ray, tomography) #mcmaster (institutions, science, institutes, university) #mskcc (organizations, hospitals, institutions, family)
Not found	#aml, #antithrombotics, #chemo, #biobusiness, #esmo, #mucmed

The following figure (Figure 24) shows how many WordNet entries have been associated to each of the 930 annotated hashtags. It may be seen that almost 600 hashtags, including the 409 hashtags that have been found directly in WordNet, are linked to a single WordNet entry (that may contain a list of synsets, if the hashtag has different meanings). The rest of the hashtags have been annotated with the support of Wikipedia. Almost 100 of them have 2 entries, and very few hashtags have more than five associated WordNet entries.

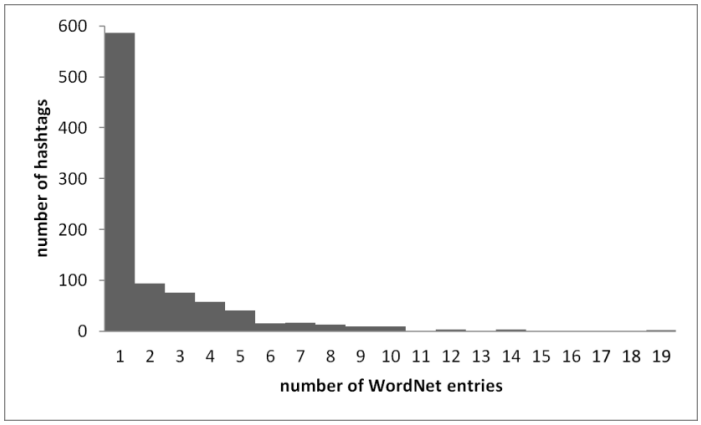


Figure 24 Distribution of WordNet entries per hashtag

In the second stage, standard bottom-up and top-down hierarchical clustering procedures (using complete linkage[13]) were applied to the set of 930 annotated hashtags, as described in section 6.2.2. The Wu-Palmer semantic similarity function (Z. Wu & Palmer, 1994) (see section 2.3.4) which is simple, easy to calculate and has been used in many semantic studies, was applied to estimate the alikeness of two WordNet synsets. This measure considers the number of nodes between the synsets in the taxonomic hierarchy, but it also takes into account their generality (concepts that appear in the upper levels of the taxonomy are considered less similar than concepts that are separated by the same distance but appear in lower levels of the taxonomy)

Finally, the most relevant clusters were selected using the mechanisms explained in section 6.2.3. The tree was analysed in a bottom-up fashion, from the horizontal cut that divides the set of hashtags in 200 clusters up to the one that partitions the set in 5 clusters. A top-down analysis from the cut in 5 clusters to the cut in 200 clusters was also performed. Finally, a full hierarchical analysis was also made.

In order to analyse the influence of the values of the filtering parameters, the selection procedures were applied 50 times, considering the following values for them:

[13] http://stat.ethz.ch/R-manual/R-patched/library/stats/html/hclust.html. Last access: November 10th, 2014

- Threshold *t1* (minimum inter-class homogeneity): values ranging from 0.5 to 0.9 in 0.1 intervals (5 values).
- Threshold *t2* (minimum number of elements): even values ranging from 2 to 20 (10 values).

As an illustrative example, the following table shows the results obtained in one of the 50 tests (with *t1*=0.7 and *t2*=10, bottom-up analysis). In this case 36 clusters were obtained. Table 23 shows for each cluster its semantic centroid, its homogeneity and its size (number of hashtags).

Table 23 Clusters obtained with t1=0.7 and t2=10

Id	Homog.	Elems	Centroid	IdClust	Homog.	Elems	Centroid
1	0,719	10	cost	19	0,962	12	s
2	0,711	11	placental	20	0,936	10	biotechnology
3	0,776	11	woody_plant	21	0,819	10	receptor
4	0,879	11	position	22	0,888	23	health
5	0,774	10	day	23	0,852	18	science
6	0,733	17	word	24	0,891	43	medicine
7	0,725	12	higher_cognitive_process	25	0,982	10	system
8	0,722	12	substance	26	0,967	10	data
9	0,902	20	therapy	27	0,965	24	people
10	0,810	17	medicine	28	0,973	11	region
11	0,892	13	state	29	0,869	29	investigation
12	0,833	12	commercial_enterprise	30	0,924	53	institution
13	0,893	13	record	31	0,819	16	activity
14	0,936	46	cancer	32	0,859	10	teaching
15	0,873	10	album	33	0,794	16	person
16	0,785	14	court	34	0,971	13	name
17	0,912	25	business	35	0,879	10	center
18	0,844	12	family	36	0,948	15	doctor

6.3.3 Evaluation

The clusters returned by the system have been evaluated with respect to a golden standard that has been manually built (section 6.3.3.1). The evaluation has been carried out from two perspectives (section 6.3.3.2): a *global* one, in which percentages of relevant clusters and identified ideal classes are calculated, and a *local* one, in which a more detailed analysis of the content of the clusters that match with ideal classes has been made.

6.3.3.1 Golden standard

A golden standard was built to evaluate the quality of the sets of hashtags obtained in each of the 50 tests. A manual analysis of the set of the annotated hashtags showed that 57.6% of them (536) were relevant medical tags, which were classified in a set *C* of 18 manually labelled categories (parts of the body, professions, medical tests, branches of Medicine, etc.). This classification is certainly subjective, but it provides a reference point with which to compare the results of the automatic clustering system. The remaining 394 hashtags (42.4%, a very high percentage) was listed as either noise or unrelated to Medicine.

Table 24 shows the following data from each of the 18 classes that have been manually detected: name (assigned manually by the human classifier), number of hashtags, inter-class homogeneity and some members of each class. It can be noticed that the homogeneity is relatively stable, with most of the values lying between 0.5 and 0.7, whereas the number of elements has a wider variety, comprising small classes with less than 10 elements and bigger classes with as many as 60 elements.

Table 24 Golden standard

Manual Class	#nhash	Hom.	Hashtags
Substances, drugs	17	0.455	cannabis, cigarettes, coffee, curcumin, marijuana, …
Medications	57	0.467	Abbott, abraxane, amgen, antibiotics, aranesp…
Biological	41	0.469	Adiponectin, biobehavioral, bioethics, biology, biomarker…
Conferences	23	0.493	amp 2012, asco 13, ash 12, asrm 2012, chest 2012…
Technologies, computer science	14	0.555	Cloud, computer, infotech, it solutions, pdf…
Medical Jobs	51	0.559	business intelligence manager, caregivers, clinician, dietician, doctor…
Temporal	21	0.573	2nd semester, 30 days of thankfulness, 365 days

			of java, annual, ashp midyear…
Academic, Research	28	0.586	clinical research , graduation, grand rounds, grants, higher ed…
Diagnosis, Symptoms	28	0.586	brain hurts, cholesterol, compassion fatigue, complications, depression…
Hospitals	14	0.624	center, cleveland clinic, clinic, department, hospice…
Body Parts	34	0.627	bladder, blood, bone, bone marrow, brain…
Cancer	60	0.634	beat cancer, bladder cancer, bowel cancer, brain cancer, breast cancer…
Clinical trials	6	0.657	adaptive trials, clinical trial, clinical trials, patient enrollment oncology clinical trials, trial…
Medical tests, treatments	41	0.659	Acupuncture, analytics, art therapy, biopsy, brachytherapy…
Health Care	22	0.664	breast health, digital health, e health, health economics, health innovation…
Geographical Locations	44	0.690	africa, alberta, arkansas, ascou, asturias…
Medical Fields	41	0.746	academic medicine, acute surgery, cardiology, emergency medicine, epidemiology…
Illness	5	0.749	acromegalia, diabetes, disease, diseases, hepatitis

6.3.3.2 Evaluation measures

This section analyses the performance of the clustering and filtering algorithm, by comparing the manual classification used as a golden standard (from now on, "classes") with the groups of hashtags selected by the system (from now on, "clusters"). Note that the 18 classes contain a partition of the 536 relevant hashtags, whereas the clusters contain a subset of the 930 annotated hashtags (and more than 40% of them were irrelevant). Thus, clusters may be very noisy. In order to evaluate the influence of the values of the two thresholds (minimum inter-class homogeneity and number of elements) the results of the 50 tests are presented.

The notation used in this section is the following:

- C is the golden standard (i.e. the set of 18 manually labelled classes of relevant hashtags).
- K is the set of clusters given as a result by the clustering and filtering procedure (i.e. a partition of a subset of the whole set of hashtags).
- Given a class c and a cluster k, a_{ck} is the number of hashtags of class c that appear in cluster k.

- *B* is a list of the 394 irrelevant hashtags.
- Given a positive integer *k*, with *k*<=|*K*|, a_{Bk} is the number of items of the list *B* in cluster *k* (i.e., the number of irrelevant hashtags in cluster number *k*).
- *N* is the number of relevant hashtags (in this case, 536).

6.3.3.2.1 Global analysis

First, it is analysed to which degree the system is able to discover the 18 relevant classes of hashtags (how many of them are found, and how many of their hashtags are correctly discovered). In this analysis the following definitions of Precision, Recall and F-measure (Equations 12, 16 and 19) are used:

Precision: This measure evaluates the percentage of clusters found by the system that are relevant.

$$PrecisionG(C,K) = \frac{|K'(C,K)|}{|K|} \tag{12}$$

$$K'(C,K) = K - noise(C,K) \tag{13}$$

$$noise(C,K) = \bigcup_{k \in K} noise(C,k) \tag{14}$$

$$noise(C,k) = \begin{cases} k, & if \max_{c \in C}\{a_{ck}\} \leq a_{Bk} \\ \emptyset, & otherwise \end{cases} \tag{15}$$

A cluster is said to be *noisy* if it shares more elements with the list B of irrelevant hashtags than with any of the 18 manually defined classes. The function *noise(C,K)* returns a list of the noisy clusters, and *K'(C,K)* is the subset of K that contains the relevant (non-noisy) clusters. To simplify the notation, this subset will be called simply *K'* in the remainder of this section.

Recall: This measure evaluates the percentage of the relevant classes that have been found by the system.

$$RecallG(C,K') = \frac{|I(C,K')|}{|C|} \tag{16}$$

$$I(C,K') = \bigcup_{k \in K'} Cmax(C,k) \tag{17}$$

$$Cmax(C,k) = \{c \in C \; such \; that \; a_{ck} = \max_{c' \in C}\{a_{c'k}\}\} \tag{18}$$

Given a non-noisy cluster *k*, *Cmax(C,k)* returns the most similar class of the golden standard *C* (i.e., the one with which it shares more elements). If there is a tie between several classes, all of them are returned. *I(C,K')* returns a list of the

classes of *C* that have been linked to at least one of the non-noisy clusters in *K'*; thus, this is the list of classes that have been discovered by the system.

F-measure: This measure is the harmonic mean of precision and recall and it evaluates the overall accuracy of the method (how many clusters are relevant and how many classes have been discovered by the system).

$$F\text{-}m(C,K) = \frac{2 * RecallG(C,K') * PrecisionG(C,K)}{RecallG(C,K') + PrecisionG(C,K)} \quad (19)$$

The following figure (Figure 25) shows the evolution of the F-Measure in the 50 tests using the bottom-up filtering procedure. The x-axis represents the value of the threshold *t1* (minimum inter-class homogeneity). The y-axis is the value of the F-measure. A coloured line is used to show the results obtained for a given value of the threshold *t2* (minimum number of elements).

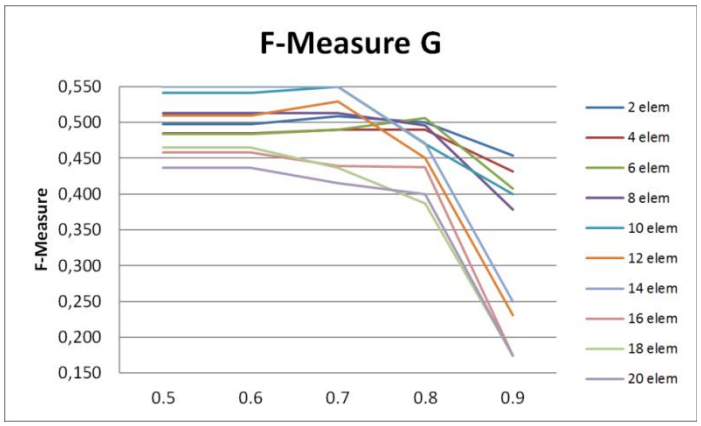


Figure 25 Global evaluation (bottom-up)

The global F-measure has the same values when t1 is 0.5 or 0.6. The best homogeneity depends on the minimum number of elements:

- For a low value of *t2* (2-6) the best homogeneity is 0.8, giving an F-measure around 0.50.
- For a medium value of *t2* (8-12) the best results (F-measures between 0.50 and 0.55) are obtained with a minimum homogeneity of 0.7.
- For higher values of *t2* the homogeneities with a higher F-value (around 0.45) are 0.5-0.6.

In all cases the results drop sharply when a minimum homogeneity of 0.9 is required. This requirement is so strong that the number of obtained clusters is very low (especially if it is also required that the clusters contain at least 10 hashtags). Remember that the actual homogeneity of the manually labelled classes of the golden standard lies mostly between 0.5 and 0.7 (only two classes exceed this value), so it is unreasonable to expect that the clusters found by the system are going to have a higher homogeneity than the "ideal" ones.

The best overall value of the F-measure (0.55) is obtained when t1=0.7 and t2=10. The results obtained with this setting were shown in Table 23. In this case 36 classes were obtained, and 16 of them were linked to 13 of the 18 golden standard classes, for a precision of 0.44 (16/36) and a recall of 0.72 (13/18). Table 25 depicts this example showing for each cluster its homogeneity, number of elements, precision and recall, automatic semantic centroid and the matching class. It may be seen that clusters 8, 25 and 35 match the same ideal class (Body Parts), whereas clusters 33 and 36 match with the class Medical Jobs. Thus, there has been a certain level of fragmentation of those two ideal classes in different clusters.

Table 25 Filtered clusters obtained with t1=0.7 and t2=10 (bottom-up)

id	Homog.	elems	Precision	Recall	centroid	Manual
3	77,58%	11	63,64%	41,18%	Woody_plant	Substances, drugs
5	77,36%	10	50,00%	23,81%	Day	Temporal
8	72,25%	12	41,67%	14,71%	Substance	Body Parts
9	90,20%	20	75,00%	36,59%	Therapy	Medical tests and treat-ments
10	80,97%	17	76,47%	22,81%	Medicine	Medications
14	93,56%	46	80,43%	61,67%	Cancer	Cancer
16	78,51%	14	42,86%	42,86%	Court	Hospitals
20	93,57%	10	60,00%	14,63%	Biotechnology	Biological
22	88,76%	23	43,48%	45,45%	Health	Health Care
24	89,10%	43	60,47%	63,41%	Medicine	Medical Fields
25	98,18%	10	70,00%	20,59%	System	Body Parts
28	97,33%	11	72,73%	18,18%	Region	Geographical Locations
32	85,87%	10	40,00%	14,29%	Teaching	Academic, Research
33	79,41%	16	37,50%	11,76%	Person	Medical Jobs
35	87,92%	10	40,00%	11,76%	Center	Body Parts
36	94,82%	15	60,00%	17,65%	Doctor	Medical Jobs
others	-	-	-	-	-	NOISE

Figure 26 shows the evolution of the F-Measure in the 50 tests using the top-down filtering procedure. The global F-measure has similar values when t1 is 0.5 or 0.6. Notice that the behaviour is the opposite of the bottom-up approach and that the minimum inter-class homogeneity (*t1*) has more impact on the best results than the minimum number of elements (*t2*).

- For a low value of *t1* (0.5-0.6) *t2* has a weak influence on the Global F-measure that is around 0.1-0.15.
- For a medium value of *t1* (0.7-0.8) the best results (F-measures between 0.4 and 0.5) are obtained with *t2* in the range 2-10.

For higher values of *t1* (above 0.8) the F-value decreases, especially when clusters are composed by 12-20 elements. The main reason is that in this dataset there are very few classes satisfying both conditions (more than 12 elements and a extremely high inter-homogeneity).

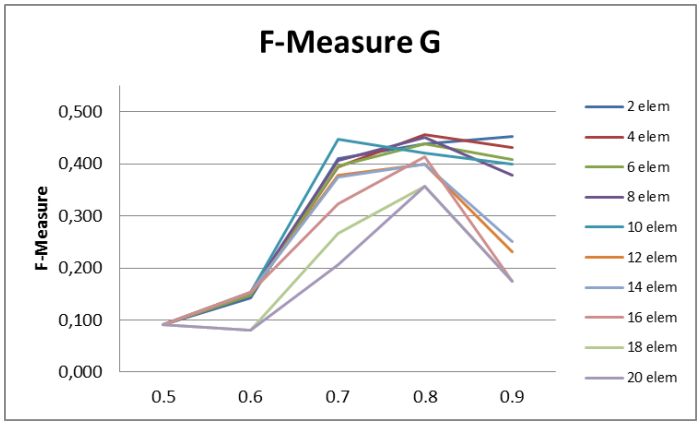


Figure 26 Global evaluation (top-down)

Setting the same thresholds that gave the best result in the bottom-up analysis (*t1*=0.7 and *t2*=10), the value of the F-measure in the top-down analysis is 0.44. In this case 20 classes were obtained, and 9 of them were linked to 8 of the 18 golden standard classes, for a precision of 0.45 (9/20) and a recall of 0.44 (8/18). Table 26 shows for each cluster its homogeneity, number of elements, precision and recall, automatic semantic centroid and the matching class. It may be seen that clusters 1 and 18 match the same ideal class (Body Parts). Thus, there has been a fragmentation of that ideal class in two different clusters like in the bottom-up analysis.

Table 26 Clusters obtained with t1=0.7 and t2=10 (top-down selection)

id	Homog.	elems	Precision	Recall	centroid	Manual
1	70,03%	17	41,18%	20,59%	organ	Body Parts
2	79,33%	18	72,22%	22,81%	medicine	Medications
12	71,42%	60	53,33%	62,75%	doctor	Medical Jobs
13	77,58%	11	63,64%	41,18%	woody_plant	Substances, drugs
15	77,36%	10	50,00%	23,81%	day	Temporal
3	72,58%	46	39,13%	40,91%	land	Geographical Locations
18	72,25%	12	41,67%	14,71%	substance	Body Parts
20	76,60%	15	40,00%	42,86%	court	Hospitals
9	77,31%	84	33,33%	68,29%	life_science	Medical Fields
others	-	-	-	-	-	NOISE

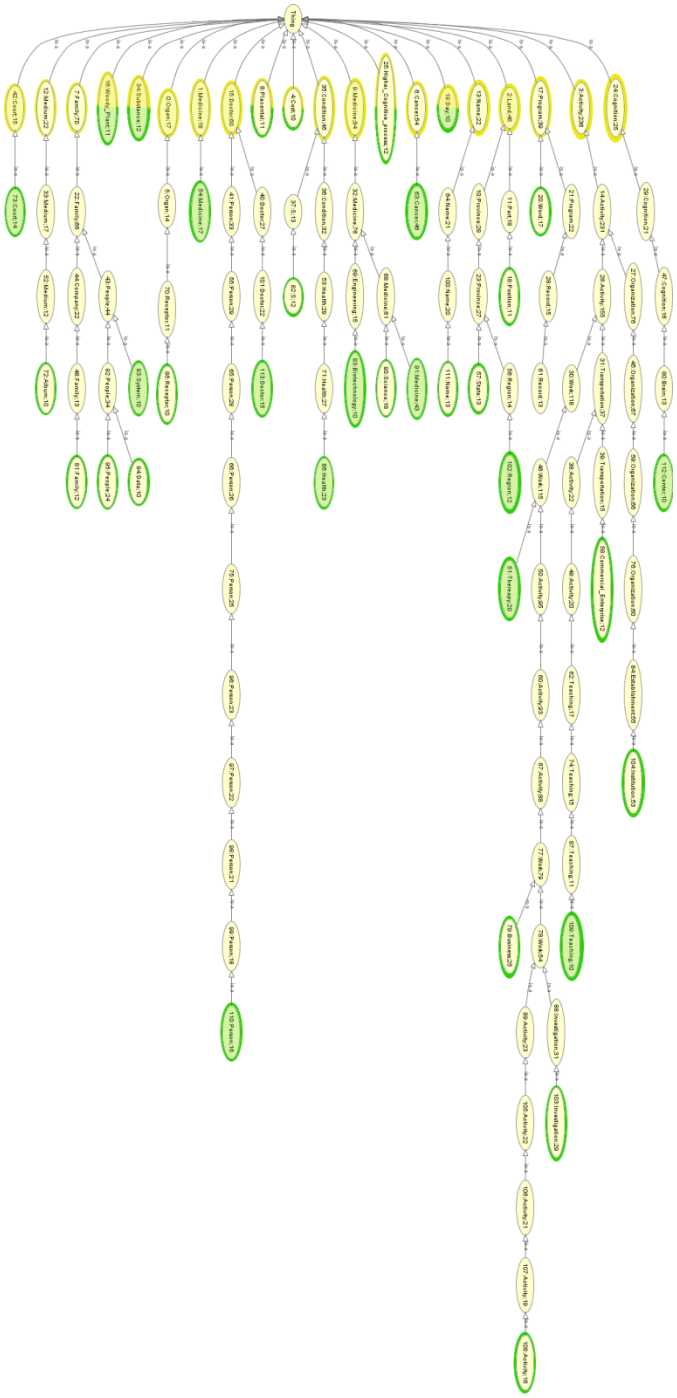


Figure 27 Final hierarchy (top-down + bottom-up approach)

Finally, Figure 27 depicts the resulting hierarchy after applying the filtering process to the full dendrogram. Each cluster is represented with a node that contains an identifier, the semantic centroid of the hashtags of the cluster and its number of elements.The nodes of the first level (with a yellow frame) represent the clusters selected by the top-down filtering approach, whereas the leaves of the tree (with a green frame) would be those selected by the bottom-up approach. The clusters fully coloured in yellow and/or green are the ones selected in one of the two selection processes that have an appropriate matching with the manually labelled classes. It may be seen how, in general, in each branch of the tree the semantic centroids go from general concepts to more specific or specialised ones (e.g., from Cognition to Brain, or from Activity to Teaching/Therapy/Organization). There are some long branches (e.g. Activity, Person) that are not very representative, as in these cases the difference between one class and its superclass is the loss of a small number of hashtags (for instance, there are Person clusters with 28, 26, 25, 23, 22 and 21 hashtags).

6.3.3.2.2 Local analysis

On a second phase of the evaluation, the quality of the clusters found by the system is studied, i.e. how well they match with the classes defined in the golden standard. In this section three different measures are considered (Equations 20, 24 and 25):

- *Precision*, *recall* and *F-measure*, defined as follows:

$$F\text{-}m(C,K') = \frac{1}{|C|}\sum_{c\in C}\max_{k\in K'}\{F(c,k)\} \tag{20}$$

$$F(c,k) = \frac{2 * Recall(c,k) * Precision(c,k)}{Recall(c,k) + Precision(c,k)} \tag{21}$$

$$Recall(c,k) = \frac{a_{ck}}{|c|} \tag{22}$$

$$Precision(c,k) = \frac{a_{ck}}{|k|} \tag{23}$$

The *precision* of a cluster *k* with respect to a class *c* (Eq. 23) is the percentage of cluster elements that belong to the class. Thus, it measures if the cluster found by the system contains items unrelated to the class. Conversely, the *recall* (Eq. 22) is the percentage of class elements that appear in the cluster, i.e. it is a measure of the coverage of the discovery system. The *F-measure* (Eq. 21) is the harmonic mean of precision and recall. The overall F-measure (Eq. 20) is the average value of the best F-measure for each of the golden standard classes.

- *Greedy many-to–one* (Zhao & Karypis, 2002):

$$GM1(C, K') = \frac{1}{N} \sum_{k \in K'} \max_{c \in C} \{a_{ck}\} \tag{24}$$

This measure is an indicator of the percentage of relevant hashtags that have been correctly assigned by the system to any of the clusters that match with one of the golden standard classes (i.e., the non-noisy clusters K').

- *F-measure tailored to multi-class clustering evaluation* (B. C. M. Fung, Wang, & Ester, 2003):

$$F\text{-}mt(C, K') = \sum_{c \in C} \frac{|c|}{N} \max_{k \in K'} \{F(c, k)\} \tag{25}$$

This measure is a weighted sum of the best F-measure for each of the golden standard classes. The weight depends on the number of elements of each class.

The following figures (Figure 28, Figure 29 and Figure 30) show the results obtained with these three measures in the 50 tests made on this case study (5 values for the minimum inter-class homogeneity, from 0.5 to 0.9, and 10 values for the minimum number of elements in each cluster, from 2 to 20), using the bottom-up filtering process). The x-axis shows the minimum homogeneity, the y-axis shows the value of the evaluation measure, and there is a different coloured line for each of the values for the minimum number of elements in each cluster.

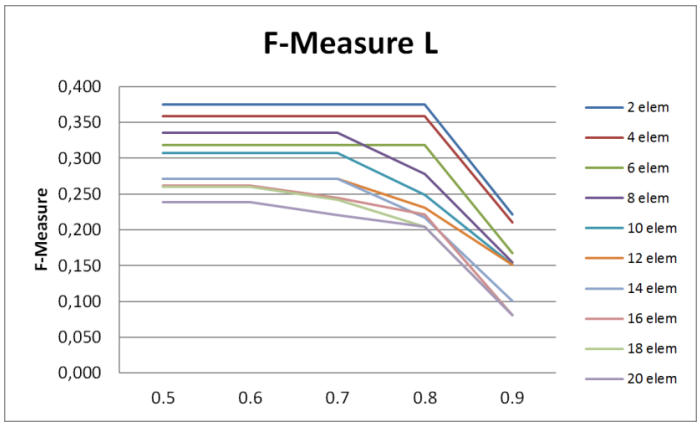


Figure 28 Study of the local F-measure (bottom-up)

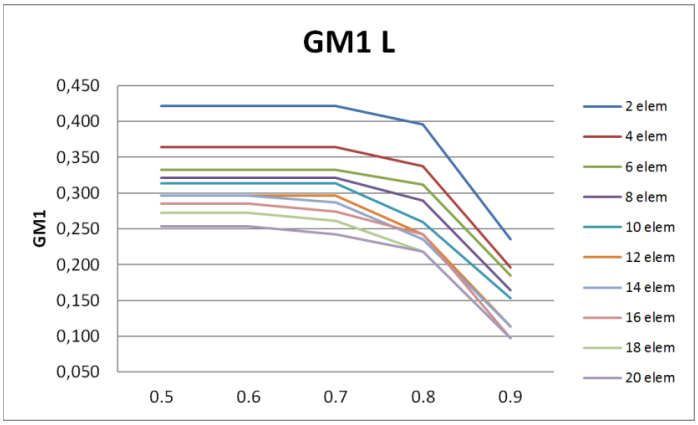


Figure 29 Study of the Greedy Many-To-One measure (bottom-up)

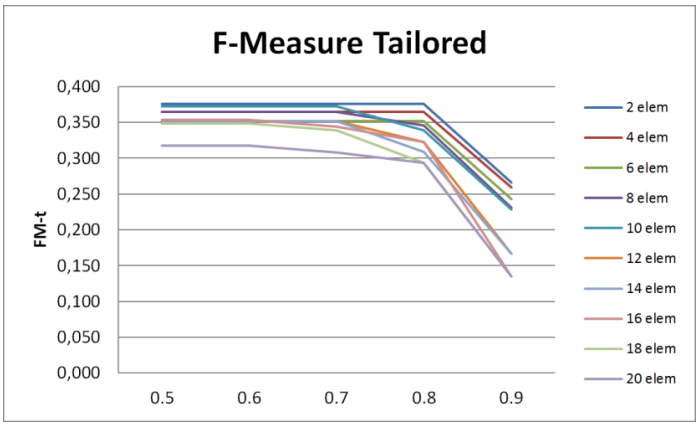


Figure 30 Study of F-Measure tailored to multi-class clustering (bottom-up)

It can be observed that the three figures show very similar results. In all the cases the values are quite stable with the smaller homogeneities (0.5 to 0.7), they decrease in 0.8 and they drop sharply in 0.9. Thus, in this case study it is reasonable to request a minimum homogeneity within each cluster of 0.7, but the quality of the results will decrease if a higher homogeneity (in fact, higher than the one of the golden standard) is forced. Concerning the second threshold, it may be clearly seen that the higher is the minimum number of elements required to select a cluster, the worse are the results. This influence is especially clear in the Greedy Many-to-One and standard F-measures. If the contribution of each class depends on its number of elements (F-measure tailored) the differences are not important until a very high homogeneity is also considered.

Figure 31, Figure 32 and Figure 33 report the results of the same experiment when the top-down filtering process is used. The three figures also show very similar results but the influence of the first threshold is more noticeable in this example. The higher the minimum inter-cluster-homogeneity (up to 0.8), the better the performance is. The reason is that this approach is not as restrictive as

the other one. In this case, the most general cluster that satisfies the threshold requirements is selected, whereas in a bottom-up approach, as soon as a small specific cluster satisfies the requirements it is selected. However, when the required homogeneity exceeds 0.8, the results decrease heavily (remember that the homogeneity of the manually defined classes was around 0.7-0.8). Concerning the second threshold, it may be seen that the quality of the results decreases when the minimum size of the clusters grows. However, it should be noted that probably the user would not be very interested in having very small clusters. If a minimum of 10 elements is considered, the best measures range between 0.230 and 0.270, which is a worse result than the one obtained with the bottom-up analysis.

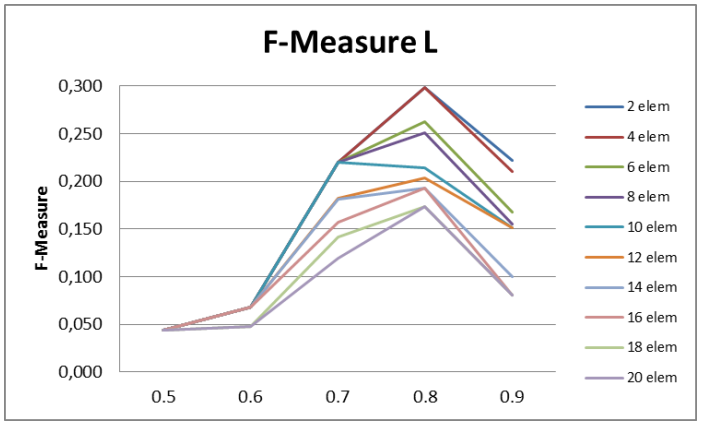


Figure 31 Study of the local F-measure (top-down)

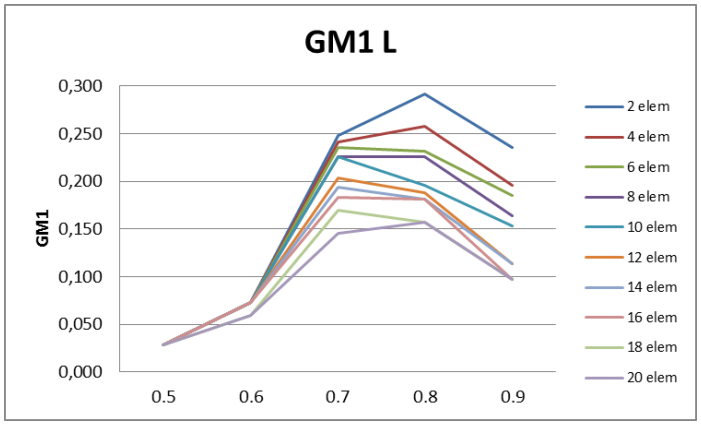


Figure 32 Study of the Greedy Many-To-One measure (top-down)

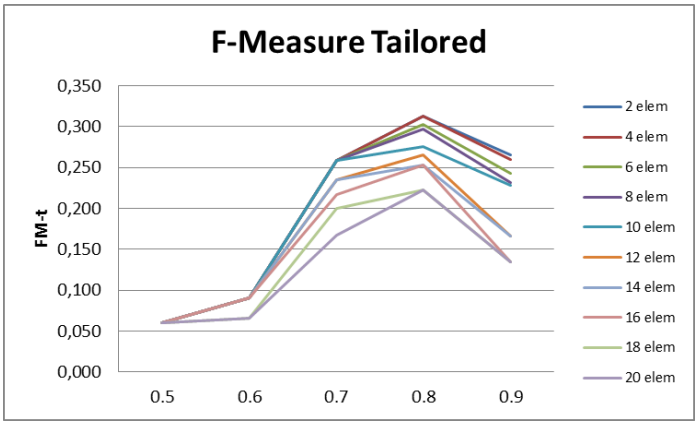


Figure 33 Study of F-Measure tailored to multi-class clustering (top-down)

In summary, the main conclusions that may be drawn from the evaluation of the Oncology case study are the following:

- The bottom-up approach shows a better overall performance than the top-down one, since it always selects the most general (and bigger) cluster from all the ones that exceed the given thresholds, and not the most specific (and smaller) one.
- Both the global and the local analysis seem to indicate that the maximum inter-cluster homogeneity that should be expected in this problem is around 0.7. This result is coherent with the fact that most of the classes of the golden standard do not exceed this value. In a situation in which the division between classes was clearer, a higher homogeneity could be considered.
- A low minimum number of elements in each cluster leads to an improvement of the local results. However, it would not be interesting to obtain as a result a set of very small clusters. Thus, the best result should present a compromise between homogeneity and size. The global results show that, in this case study, the best combination of homogeneity, size, precision and recall is obtained for clusters that contain at least 10 elements and have at least a 0.7 homogeneity.

6.4 Summary

The main disadvantage of most of the current approaches to automatic topic detection in Twitter is that they make a purely syntactic analysis of the content of the tweets, relying on the frequent co-ocurrence of related terms. The new proposal presented in this chapter goes a step further, advocating for a semantic

treatment of the hashtags that considers properly their meaning in the clustering process.

In the first step of semantic annotation, hashtags are linked to WordNet entries, directly or through the analysis of Wikipedia categories (section 6.2.1). Some interesting properties of this first step are the following:

- It permits to move from the syntactic appearance of a word to the semantic level, as each WordNet entry has a list of synsets that describe the possible meanings of the word.
- It permits to consider general concepts (that are found directly in WordNet) as well as acronyms and specific named entities (which are mostly found in Wikipedia, although some of them also appear in WordNet).
- As the system does not apply any stemming procedure, it is possible to find different meanings for lexically similar words (e.g. 'pharma', 'pharmaceutical' and 'pharmacology' have different WordNet entries).
- The use of word-breaking techniques permits to deal with multi-word hashtags such as #LungCancer or #HighBloodPressure. The search of rightmost substrings in WordNet permits to detect the most general concept that matches with the hashtag.
- Synonyms are automatically linked to the same concept (e.g. 'illness', 'malady' and 'sickness' have a shared synset in WordNet).
- Meaningless expressions or invented words will not be found either in WordNet or Wikipedia, so they will be filtered out in this first stage of the analysis.

One aspect that could be improved in this first step is the analysis of situations in which a single hashtag (e.g. a short acronym) has different Wikipedia entries. For example, #ESMO might refer to the European Society for Medical Oncology or to the European Student Moon Orbiter. Right now the system does not know how to choose the correct option, and the hashtag is not annotated. In the future work it could be possible to introduce a disambiguation procedure, for instance by searching for other hashtags of the set within the content of the Wikipedia articles associated to the available options. In the Oncology case study it should not be too hard to discover that the first option for ESMO is the most appropriate one (in fact, #Oncology is the most frequent hashtag in the set), but the situation could be much more complex (e.g., Wikipedia offers 14 different possibilities for #AML).

In the second step of the analysis, a semantic hierarchical clustering of the annotated hashtags is performed (section 6.2.2). Some relevant aspects of this step are the following:

- At this point every hashtag has been linked to one (or several) WordNet entries, each of which may contain a list of synsets.

Therefore, a hashtag is represented by a list of possible concepts. The similarity between hashtags is calculated at a semantic level, applying an ontology-based semantic similarity measure on these lists; thus, in this clustering step the initial syntactic form of the hashtag is no longer taken into account. The idea is that a semantic procedure of this nature will improve the quality of the resulting hashtags (with respect to the methods based on syntactic co-occurrence). For instance, synonym hashtags will certainly have a very strong similarity, as they will appear together in a synset.

- Notice that, as commented in section 6.3.1, most of the previous works on hashtag clustering use the frequency of the co-occurrence of two hashtags as their degree of similarity. As tweets do not usually have long lists of hashtags, the co-occurrence matrix is very sparse. One of the main advantages of the semantic approach proposed in this work is that most pairs of hashtags will have a non-null similarity, because their associated concepts may have some kind of relationship, even if it is small. Thus, it will be possible to make a much more fine-grained analysis of the relationships between hashtags.

A possible line of future work at this stage is the specific treatment of polysemic hashtags (those linked to WordNet entries with several synsets, like #cancer). Now the system considers that the similarity between two hashtags is the maximum similarity between their possible senses. This option seems appropriate when the set of tweets that is being analysed belongs to a certain domain of discourse, as the correct sense of polysemic words will be uncovered. However, if the system was fed with a general unfiltered set of tweets, it could be the case that clusters that merge different topics could be generated. For instance, a set like {#Sagittarius, #Cancer, #LungCancer} could emerge, because Cancer has different senses with a high similarity with the other two elements. One possibility to tackle these situations could be to analyse which is the synset that matches with other hashtags. In the example, it could be possible to note that the Cancer synsets that match with Sagittarius and LungCancer are different, and then generate two different clusters {#Sagittarius, #Cancer-1}, {#Cancer-2, #LungCancer}. In any case, it should be noted that WordNet synsets are quite subtle, so one should be careful when deciding to split a hashtag in different clusters.

Finally, in the last step a bottom-up dynamic analysis of the hierarchical tree of clusters is made, trying to find out which are the clusters that are worth keeping (those that represent semantically coherent topics that are covered by a sufficiently large number of hashtags). The current procedure finds a set of disjoint clusters, which are the most specific (or the most general) ones that satisfy the minimum homogeneity and size requirements. By making a full analysis of the dendrogram and keeping all the clusters that satisfy the given constraints, the system is also able to obtain a tree of topics of different levels of generality, rather than a list of disjoint topics. A deeper study of this procedure, in which long branches of very similar clusters are avoided, might be a subject of future work.

In summary, the main idea of this chapter is that topic detection procedures focused on the analysis of hashtags should move from co-occurrence measures towards a semantic understanding and treatment of the terms appearing on the hashtags. The method proposed in this dissertation is a first step in this direction, although many lines of work are currently open.

The main publications related to the contributions presented in this chapter are the following:

Vicient, C., Moreno, A. (2014). Unsupervised topic discovery in micro-blogging networks. Expert Systems with Applications (under review).

Vicient, C., Moreno, A. (2013). A Study on the Influence of Semantics on the Analysis of Micro-blog Tags in the Medical Domain. (A. Cuzzocrea, C. Kittl, D. E. Simos, E. Weippl, & L. Xu, Eds.) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 8127, pp. 446–459.

Vicient, C., Moreno, A. (2014) Unsupervised semantic clustering of Twitter hashtags. Proceedings of the 21st European Conference on Artificial Intelligence (ECAI-2014), Eds. T.Schaub et al., pp. 1119-1120. Prague, Czech Republic, August 2014.

Chapter 7
Conclusions and future work

The overwhelming amount of electronic data accessible via Internet, caused by the proliferation of social technologies, has sparked among the research community a great interest on the development of tools that facilitate their acquisition, exploitation and consumption. The scientists working in the *Knowledge Discovery* field, which constitutes the starting point for these areas of study, have already made many important contributions in the last years. However, there are two generalised problems concerning the works proposed in the literature. On the one hand, most of them apply traditional *Data Mining* and *Information Retrieval* approaches which rely on a) the size of the analysed dataset and b) the co-occurrence between the terms contained in the set. On the other hand, the methodologies that aim to go a step further by incorporating semantic domain knowledge usually depend on manually pre-annotated sets, which are still relatively limited. Finally, the exponential growing nature of the Web contents suggests that only unsupervised approaches may survive in the future, since the supervised and semi-supervised ones, as they need some level of human intervention, will necessarily lead to bottlenecks. So, unsupervised and domain-independent data analysis techniques able to deal with semantics and automatic annotation mechanisms are needed to support a viable transition between the current approaches and the forthcoming semantic technologies.

In this thesis, different automatic and unsupervised domain-independent tools well suited for the semantic analysis of a range of Web resources have been proposed.

The first contribution of this thesis is the definition of a general framework that enables the extraction of the most relevant features from a range of textual documents in which the representative features of the studied entity are semantically annotated with the support of a domain ontology. The proposed semantic framework has two basic steps:

- The **detection and selection** of the most representative entities which describe the analysed resource.
- The **semantic annotation** of the selected entities that, eventually, become the features that represent the studied real entity.

One of the main conclusions of this first part of the dissertation is that the development of automatic tools must be a priority to enhance the scalability and the usability of current *Information Extraction* approaches and to be able to apply new semantic methodologies to the actual (mainly textual) Web resources. It has also been proved that Named Entities (selected and filtered with Web-scale statistics) describe, in a way less ambiguous than general words, the most important characteristics of an analysed real entity. This assertion is true especially when the studied resources lack a formal structure, although it is possible to take profit of different kinds of semi-structured inputs to improve even more the detection of relevant features. In fact, the evaluation suggests that the clever use of semi-structured inputs may produce quality results with a lower computational cost. Moreover, it has been demonstrated that the use of the Web as a general learning corpus (to establish a bridge from the Named Entity level to the conceptual level) can improve the recall of the annotation regardless of the coverage limitations of Named Entities. Another important aspect concerning the performance of the proposed methodology is that its quality is proportional to the quality of the domain ontologies used to drive the extraction and annotation process; thus, the better and more accurate is the ontology, the higher precision is achieved avoiding the ambiguity of the terms in a specific domain. The main drawback of this novel Information Extraction approach, which was the detonator for the second part of the work, is that it cannot deal appropriately with Social Web sources that exhibit a reduced context, like for instance tweets.

The next contribution of this thesis is the modification of the above framework to adapt it to the limitations that social resources – with a reduced context – present, focusing the attention in micro-blogging services such as Twitter. In the previous approach, a Web document was described with a set of relevant features. Tweets are so short that they cannot be meaningfully represented in an individual way by a set of term, so a new approach was devised. The proposed procedure aims to find the hashtags present in a set of tweets, annotate them semantically and calculate all the related clusters of hashtags to figure out the main topics of the dataset. Again, the proposed semantic framework is automatic, unsupervised and domain-independent. It consists of the following steps:

- The **semantic annotation of hashtags** in which hashtags are mapped with concepts contained in the WordNet ontology.
- The **clustering of hashtags** according to the semantic similarity between their associated concepts.
- A **filtering process** in which the most relevant classes are detected and the noisy elements are dropped.

The results obtained by this approach permit to conclude that some of the problems that this sort of social networks have may be minimised by the use of semantics. The unstructured and unlimited nature of hashtags accentuates problems such as synonymy, polysemy, lack of any form of explicit organization or normalization, etc. For example, synonymous hashtags will hardly co-occur, so the standard methods based on a purely syntactic analysis of the co-occurrence

between pairs of terms will be unlikely to group synonyms in the same cluster. In the same way, polysemic hashtags are likely to appear in a single cluster whereas semantic approaches may be able to classify syntactically equivalent terms into different groups according to its meaning. For instance, if the hashtag #cancer co-occurs with the hashtag #tumour 10 times and with the hashtag #zodiac 2 times, syntactic approaches will probably cluster together #cancer, #tumour and #zodiac whereas semantic approaches could classify the polysemic hashtag into two different clusters according to their meaning or, as in the case of the method proposed in Chapter 6 will be able to detect the correct meaning of the term if all the tweets belong to a particular domain. Moreover, natural language terms are ambiguous and even with semantic information it may be hard to figure out the meaning of a hashtag. Following the previous example, it would be difficult to cluster the terms #cancer and #leo since the first one might be related with the illness or with the sign of the zodiac and the second one might represent a real person named Leo or also a zodiac sign. Regarding the clustering process, it has been shown that only roughly 16% of hashtags co-occur in a dataset so the hashtag co-occurrence matrix is usually very sparse making it difficult or impossible to compare two hashtags that never co-occur, whereas the semantic approach enables the comparison at a conceptual level of all of the hashtags. Thus, it is possible to make a much more fine-grained analysis of the relationships between hashtags.

Another important aspect of most of the current topic detection methods is that they usually present merging and fragmentation problems, and an appropriate and non-trivial setting of the thresholds is important to minimise their effects. The generality of the domain may be a relevant factor in the quality of the results. If the domain is very specific a fine-grained tuning of the parameters will be needed to try to minimise the merging effects, whereas a general domain will probably be more prone to fragmentation problems.

The contributions of this Ph.D. thesis have been applied and evaluated within the context of Tourism, cinema and medical domains, exploiting semantic structured knowledge like domain ontologies, general-purpose ontologies like WordNet and semi-structured repositories like Wikipedia, enabling a semantically coherent interpretation of the terms contained in each one of the analysed domains.

To sum up, as a general conclusion of the thesis it has been shown that the incorporation of semantic information, i.e. the analysis of the actual meaning of the terms, may help to alleviate some of the present issues of data mining methods and to make a better clustering of them, which will lead to an improved performance of new semantic methods that rely on pre-annotated inputs.

Finally, the scientific papers that have been published about the work done in this Ph.D. Thesis are the following:

Journals:

- Vicient, C., Sánchez, D., Moreno, A. (2013). *An automatic approach for ontology-based feature extraction from heterogeneous textual*

resources. Engineering Applications of Artificial Intelligence 26, pp. 1092-1106.

- Moreno, A., Valls, A., Martínez, S., Vicient, C., Marín, L., Mata, F. *Personalised recommendations based on novel semantic similarity and clustering procedures*. AI Communications. Accepted for publication, in press.
- Vicient, C., Moreno, A. (2014). *Unsupervised topic discovery in micro-blogging networks*. Expert Systems with Applications (under review).

International conferences:

- Vicient, C., Sánchez, D., Moreno, A. (2011). *Ontology-Based Feature Extraction*. In Workshop on 4th Natural Language Processing and Ontology Engineering (NLPOE 2011) in IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT 2011) (Vol. 3, pp. 189–192). Lyon (France).
- Vicient, C., Sánchez, D., Moreno, A. (2011). *A Methodology to Discover Semantic Features from Textual Resources*. In Sixth International Workshop on Semantic Media Adaptation and Personalization (SMAP 2011) (pp. 39–44). Vigo (Spain)
- Moreno, A., Valls, A., Mata, F., Martínez, S., Marín, L.,Vicient, C. (2013). *A semantic similarity measure for objects described with multi-valued categorical attributes*. Series Frontiers in Artificial Intelligence and Applications (Vol. 256, pp. 263–272).
- Vicient, C., Moreno, A. (2013). *A Study on the Influence of Semantics on the Analysis of Micro-blog Tags in the Medical Domain*. (A. Cuzzocrea, C. Kittl, D. E. Simos, E. Weippl, & L. Xu, Eds.) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 8127, pp. 446–459.
- Vicient, C., Moreno, A. (2014) *Unsupervised semantic clustering of Twitter hashtags*. Proceedings of the 21st European Conference on Artificial Intelligence (ECAI-2014), Eds. T.Schaub et al., pp. 1119-1120. Prague, Czech Republic, August 2014

As further work, several research lines may be proposed:

- It is a priority to study how to reduce the number of queries (e.g. using only a subset of Hearst Patterns) to Web search engines in the Information Extraction method, as they are the slowest part of the algorithm and they introduce a dependency on external resources.
- It is also important to evaluate the behaviour and applicability of the proposed Information Extraction methodology in other domains and using other ontologies.

- Another relevant task is the collection of a large set of documents annotated by experts in order to compare the automatic annotation performed by our approach with the one made by experts.
- Another important issue is to study how to deal with the semantic ambiguity of Wikipedia entries in those situations in which a single hashtag (e.g. a short acronym) has different Wikipedia pages.
- A possible line of future work is the specific treatment of polysemic hashtags (those linked to WordNet entries with several synsets, like #cancer).
- Another interesting line of future work is the development of a recommender system that uses the topics extracted from a dataset to recommend the most interesting tweets to a user, according to his specific interests.
- Along the same line, it could also be possible to recommend to a user the hashtags that could be employed in a given tweet (for instance, those hashtags that belong to the cluster that is more semantically similar to the terms used in the tweet).

References

Abril, D., Navarro-Arribas, G., & Torra, V. (2011). On the Declassification of Confidential Documents. In V. Torra, Y. Narakawa, J. Yin, & J. Long (Eds.), *Modeling Decision for Artificial Intelligence SE - 22* (Vol. 6820, pp. 235–246). Springer Berlin Heidelberg. doi:10.1007/978-3-642-22589-5_22

Ahmad, K., Tariq, M., Vrusias, B., & Handy, C. (2003). Corpus-Based Thesaurus Construction for Image Retrieval in Specialist Domains. In F. Sebastiani (Ed.), *Advances in Information Retrieval SE - 36* (Vol. 2633, pp. 502–510). Springer Berlin Heidelberg. doi:10.1007/3-540-36618-0_36

Aiello, L. M., Petkos, G., Martin, C., Corney, D., Papadopoulos, S., Skraba, R., … Jaimes, A. (2013). Sensing Trending Topics in Twitter. *Multimedia, IEEE Transactions on*, *15*(6), 1268–1282. doi:10.1109/TMM.2013.2265080

Alfayez, R., & Joy, M. (2013). Inferring dynamic taxonomies for terms based on UGC. In *Third International Conference on Innovative Computing Technology (INTECH), 2013* (pp. 545–550). doi:10.1109/INTECH.2013.6653644

Alfonseca, E., & Manandhar, S. (2002). Improving an Ontology Refinement Method with Hyponymy Patterns. In *Proceedings of 3rd International Conference on Language Resources and Evaluation, LREC 2002*. Las Palmas, Spain. doi:10.1.1.19.7172

Archambault, D., Greene, D., & Cunningham, P. (2013). TwitterCrowds: Techniques for Exploring Topic and Sentiment in Microblogging Data. *CoRR, abs/1306.3*.

Banko, M., & Etzioni, O. (2008). The Tradeoffs Between Open and Traditional Relation Extraction. In *Proceedings of ACL-08: HLT* (pp. 28–36). Columbus, Ohio: Association for Computational Linguistics.

Batet, M. (2011). Ontology-based semantic clustering. *AI Communications*, *24*(3), 291–292. doi:10.3233/AIC-2011-0501

Batet, M., Valls, A., & Gibert, K. (2011). Semantic Clustering based on Ontologies - An Application to the Study of Visitors in a Natural Reserve. In *ICAART 2011 - Proceedings of the 3rd International Conference on Agents and Artificial Intelligence, Volume 1* (pp. 283–289). Rome, Italy.

Baumgartner, R., Flesca, S., & Gottlob, G. (2001). Visual Web Information Extraction with Lixto. In P. M. G. Apers, P. Atzeni, S. Ceri, S. Paraboschi, K. Ramamohanarao, & R. T. Snodgrass (Eds.), *27th International Conference on Very Large Data Bases (VLDB)* (pp. 119–128). Roma, Italy: Morgan Kaufmann.

Becker, H., Naaman, M., & Gravano, L. (2011). Beyond trending topics: Real-world event identification on Twitter. In *Fifth International AAAI Conference on Weblogs and Social Media*.

Berland, M., & Charniak, E. (1999). Finding Parts in Very Large Corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics* (pp. 57–64). Stroudsburg, PA, USA: Association for Computational Linguistics. doi:10.3115/1034678.1034697

Berners-Lee, T., & Hendler, J. (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.

Bhulai, S., Kampstra, P., Kooiman, L., Koole, G., Deurloo, M., & Kok, B. (2012). Trend visualization in Twitter: what's hot and what's not? In *DATA ANALYTICS 2012, The First International Conference on Data Analytics*, (pp. 43–48). Barcelona: IARIA.

Bisson, G., Nédellec, C., & Cañamero, D. (2000). Designing clustering methods for ontology building: The Mo'K Workbench. In *In Proceedings of the ECAI Ontology Learning Workshop* (pp. 13–19).

Blei, D. M., & Lafferty, J. D. (2006). Dynamic Topic Models. In *Proceedings of the 23rd International Conference on Machine Learning* (pp. 113–120). New York, NY, USA: ACM. doi:10.1145/1143844.1143859

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *The Journal of Machine Learning Research*, *3*, 993–1022.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2012). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, *3*, 993–1022. doi:10.1162/jmlr.2003.3.4-5.993

Bollegala, D., Matsuo, Y., & Ishizuka, M. (2007). Measuring semantic similarity between words using web search engines. In *WWW '07: Proceedings of the 16th international conference on World Wide Web* (pp. 757–766). New York, NY, USA: ACM. doi:http://doi.acm.org/10.1145/1242572.1242675

Boyd, D., & Crawford, K. (2012). Critical questions for big data: Provocations for a cultural, technological, and scholarly phenomenon. *Information Communication and Society*, *15*(5), 662–679.

Bremmer, C. (2007). Top 150 City Destinations: London leads the way. Retrieved from http://blog.euromonitor.com/2007/10/top-150-city- destinations-london-leads-the-way.html

Brill, E. (2003). Processing natural language without natural language processing. In *Proceedings of the 4th international conference on Computational linguistics and intelligent text processing* (pp. 360–369). Berlin, Heidelberg: Springer-Verlag.

Buitelaar, P., Cimiano, P., Frank, A., Hartung, M., & Racioppa, S. (2008). Ontology-based information extraction and integration from heterogeneous data sources. *International Journal of Human-Computer Studies*, *66*, 759–788. doi:10.1016/j.ijhcs.2008.07.007

Buitelaar, P., Olejnik, D., & Sintek, M. (2004). A Protégé Plug-In for Ontology Extraction from Text Based on Linguistic Analysis. In C. Bussler, J. Davies, D. Fensel, & R. Studer (Eds.), *The Semantic Web: Research and Applications SE - 3* (Vol. 3053, pp. 31–44). Springer Berlin Heidelberg. doi:10.1007/978-3-540-25956-5_3

Cafarella, M., Downey, D., Soderland, S., & Etzioni, O. (2005). KnowItNow: fast, scalable information extraction from the web. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing* (pp. 563–570). Vancouver, Canada: Association for Computational Linguistics. doi:10.3115/1220575.1220646

Califf, M. E., & Mooney, R. J. (2003). Bottom-up Relational Learning of Pattern Matching Rules for Information Extraction. *J. Mach. Learn. Res.*, *4*, 177–210. doi:10.1162/153244304322972685

Cantador, I., Konstas, I., & Jose, J. M. (2011). Categorising social tags to improve folksonomy-based recommendations. *Web Semantics: Science, Services and Agents on the World Wide Web*, *9*(1), 1–15. doi:10.1016/j.websem.2010.10.001

Caraballo, S. A. (1999). Automatic Construction of a Hypernym-labeled Noun Hierarchy from Text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics* (pp. 120–126). Stroudsburg, PA, USA: Association for Computational Linguistics. doi:10.3115/1034678.1034705

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, E. R., & Mitchell, T. M. (2010). Toward an architecture for never-ending language learning. In *In AAAI*.

Cataldi, M., Di Caro, L., & Schifanella, C. (2010). Emerging Topic Detection on Twitter Based on Temporal and Social Terms Evaluation. In *Proceedings of the Tenth International Workshop on Multimedia Data Mining* (pp. 4:1–4:10). New York, NY, USA: ACM. doi:10.1145/1814245.1814249

Çelik, D., & Elçi, A. (2013). An Ontology-based Information Extraction Approach for RéSumés. In *Proceedings of the 2012 International Conference on Pervasive Computing and the Networked World* (pp. 165–179). Berlin, Heidelberg: Springer-Verlag. doi:10.1007/978-3-642-37015-1_14

Celikyilmaz, A., Hakkani-Tur, D., & Feng, J. (2010). Probabilistic model-based sentiment analysis of twitter messages. In *Spoken Language Technology Workshop (SLT), 2010 IEEE* (pp. 79–84). doi:10.1109/SLT.2010.5700826

Choi, I., & Kim, M. (2003). Topic Distillation Using Hierarchy Concept Tree. In *Proceedings of the 26th Annual International ACM SIGIR (Conference on Research and Development in Informaion Retrieval)* (pp. 371–372). New York, NY, USA: ACM. doi:10.1145/860435.860506

Church, K., Gale, W., Hanks, P., & Kindle, D. (1991). Using statistics in lexical analysis. *Lexical Acquisition: Exploiting on-Line Resources to Build a Lexicon*, 115.

Cilibrasi, R. L., & Vitányi, P. M. B. (2006). The Google Similarity Distance. *IEEE Transactions on Knowledge and Data Engineering*, *19*(3), 370–383. doi:10.1109/TKDE.2007.48

Cimiano, P. (2006a). *Ontology Learning and Population from Text: Algorithms, Evaluation and Applications*. Secaucus, NJ, USA: Springer-Verlag New York, Inc.

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

Cimiano, P., Handschuh, S., & Staab, S. (2004). Towards the Self-annotating Web. In *Proceedings of the 13th International Conference on World Wide Web* (pp. 462–471). New York, NY, USA: ACM. doi:10.1145/988672.988735

Cimiano, P., Ladwig, G., & Staab, S. (2005). Gimme' the Context: Context-driven Automatic Semantic Annotation with C-PANKOW. In *Proceedings of the 14th International Conference on World Wide Web* (pp. 332–341). New York, NY, USA: ACM. doi:10.1145/1060745.1060796

Ciravegna, F., Dingli, A., Guthrie, D., & Wilks, Y. (2003). Integrating Information to Bootstrap Information Extraction from Web Sites. In *Proceedings of the {IJCAI} 2003 Workshop on Information Integration on the Web, workshop in conjunction with the 18th {International Joint Conference on Artificial Intelligence (IJCAI 2003)}*.

Ciravegna, F., Dingli, A., Petrelli, D., & Wilks, Y. (2002). User-System Cooperation in Document Annotation based on Information Extraction. In *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management, EKAW02* (pp. 122–137). Sigüenza, Spain: Springer Verlag.

Ciravegna, F., Gentile, A. L., & Zhang, Z. (2012). LODIE: Linked Open Data for Web-scale Information Extraction. In D. Maynard, M. van Erp, & B. Davis (Eds.), *SWAIE* (Vol. 925, pp. 11–22). CEUR-WS.org.

Cotelo, J. M., Cruz, F. L., & Troyano, J. A. (2014). Dynamic topic-related tweet retrieval. *JASIST*, *65*(3), 513–523.

Cunningham, H., Maynard, D., Bontcheva, K., & Tablan, V. (2002). GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics, ACL 2002* (pp. 168–175). Philadelphia, US. doi:10.3115/1073083.1073112

Daude, J., Padro, L., & Rigau, G. (2003). Validation and tuning of wordnet mapping techniques. In *Proceedings of RANLP*.

Dela Rosa, K., Shah, R., Lin, B., Gershman, A., & Frederking, R. (2011). Topical Clustering of Tweets. *Proceedings of the ACM SIGIR: SWSM*.

Deza, M., & Deza, E. (2009). Encyclopedia of Distances. In *Encyclopedia of Distances SE - 1* (pp. 1–583). Springer Berlin Heidelberg. doi:10.1007/978-3-642-00234-2_1

Dias, G., Santos, C., & Cleuziou, G. (2006). Automatic Knowledge Representation using a Graph-based Algorithm for Language-Independent Lexical Chaining. In *Proceedings of the Workshop on Information Extraction Beyond The Document* (pp. 36–47). Sydney, Australia: Association for Computational Linguistics.

Dill, S., Eiron, N., Gibson, D., Gruhl, D., Guha, R., Jhingran, A., … Zien, J. Y. (2003). A case for automated large-scale semantic annotation. *Web Semantics, Services and Agents on the World Wide Web*, *1*, 115–132. doi:10.1016/j.websem.2003.07.006

Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R. S., Peng, Y., … Sachs, J. (2004). Swoogle: A Search and Metadata Engine for the Semantic Web. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management* (pp. 652–659). New York, NY, USA: ACM. doi:10.1145/1031171.1031289

Downey, D., Broadhead, M., & Etzioni, O. (2007). Locating complex named entities in web text. In *IJCAI International Joint Conference on Artificial Intelligence* (pp. 2733–2739). doi:10.1.1.104.6523

Dreer, F., Saller, E., & Elsässer, P. (2014). Building a Social Dictionary based on Twitter Data. In *Tagung der Computerlinguistikstudierenden* (pp. 1–5).

Embley, D. W., Campbell, D. M., Smith, R. D., & Liddle, S. W. (1998). Ontology-based Extraction and Structuring of Information from Data-rich Unstructured Documents. In *Proceedings of the seventh international conference on Information and knowledge management - CIKM '98* (pp. 52–59). Bethesda, Maryland, USA. doi:10.1145/288627.288641

Etzioni, O., Banko, M., Soderland, S., & Weld, D. S. (2008). Open information extraction from the web. *Commun. ACM*, *51*(12), 68–74. doi:10.1145/1409360.1409378

Etzioni, O., Cafarella, M., Downey, D., Kok, S., Popescu, A.-M., Shaked, T., … Yates, A. (2004). Web-scale Information Extraction in KnowItAll. In *Proceedings of WWW-2004*.

Etzioni, O., Cafarella, M., Downey, D., Popescu, A.-M., Shaked, T., Soderland, S., … Yates, A. (2005). Unsupervised Named-Entity Extraction from the Web: An Experimental Study. *ARTIFICIAL INTELLIGENCE*, *165*, 91–134.

Etzioni, O., Fader, A., Christensen, J., Soderland, S., & Mausam, M. (2011). Open Information Extraction: The Second Generation. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume One* (pp. 3–10). AAAI Press. doi:10.5591/978-1-57735-516-8/IJCAI11-012

Everitt, B. S., Landau, S., & Leese, M. (2001). *Cluster analysis* (Fourth ed.). London: Arnold.

Fader, A., Soderland, S., & Etzioni, O. (2011). Identifying Relations for Open Information Extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 1535–1545). Stroudsburg, PA, USA: Association for Computational Linguistics.

Fang, Y., Zhang, H., Ye, Y., & Li, X. (2014). Detecting hot topics from Twitter: A multiview approach. *Journal of Information Science*, *40*(5), 578–593.

Faria, C., Serra, I., & Girardi, R. (2014). A domain-independent process for automatic ontology population from text. *Science of Computer Programming*, *95, Part 1*(0), 26–43. doi:http://dx.doi.org/10.1016/j.scico.2013.12.005

Fayyad, U., Piatetsky-shapiro, G., & Smyth, P. (1996). From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, *17*, 37–54.

Feilmayr, C., Parzer, S., & Pröll, B. (2009). Ontology-Based Information Extraction from Tourism Websites. *Journal of Information Technology & Tourism*, *11*(3), 183–196. doi:10.3727/109830509X12596187863874

Fellbaum, C. (Ed.). (1998). *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. MIT Press.

Ferrara, E., JafariAsbagh, M., Varol, O., Qazvinian, V., Menczer, F., & Flammini, A. (2013). Clustering memes in social media. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining - ASONAM '13* (pp. 548–555). doi:10.1145/2492517.2492530

Fleischman, M., & Hovy, E. (2002). Fine Grained Classification of Named Entities. *Proceedings of the 19th International Conference on Computational Linguistics*, *1*, 1–7. doi:10.3115/1072228.1072358

Flesca, S., Manco, G., Masciari, E., Rende, E., & Tagarelli, A. (2004). Web Wrapper Induction: A Brief Survey. *AI Commun.*, *17*(2), 57–61.

Forgy, E. (1965). Cluster analysis of multivariate data: efficiency versus interpretability of classifications. *Biometrics*, *21*, 768–769.

Freedman, M., Ramshaw, L., Boschee, E., Gabbard, R., Kratkiewicz, G., Ward, N., & Weischedel, R. (2011). Extreme Extraction: Machine Reading in a Week. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 1437–1446). Stroudsburg, PA, USA: Association for Computational Linguistics.

Freitag, D. (1998). Toward General-purpose Learning for Information Extraction. In *Proceedings of the 17th International Conference on Computational Linguistics - Volume 1* (pp. 404–408). Stroudsburg, PA, USA: Association for Computational Linguistics. doi:10.3115/980451.980914

Freitag, D., & Mccallum, A. K. (1999). Information Extraction with HMMs and Shrinkage. In *In Proceedings of the AAAI-99 Workshop on Machine Learning for Information Extraction* (pp. 31–36).

Fuentes-Lorenzo, D., Fernández, N., Fisteus, J. A., & Sánchez, L. (2013). Improving large-scale search engines with semantic annotations. *Expert Systems with Applications*, *40*(6), 2287–2296. doi:http://dx.doi.org/10.1016/j.eswa.2012.10.042

Fung, B. C. M., Wang, K., & Ester, M. (2003). Hierarchical Document Clustering Using Frequent Itemsets. In *Proceedings of the SIAM International Conference on Data Mining* (Vol. 30, pp. 59–70).

Fung, G. P. C., Yu, J. X., Yu, P. S., & Lu, H. (2005). Parameter Free Bursty Events Detection in Text Streams. In *Proceedings of the 31st International Conference on Very Large Data Bases* (pp. 181–192). Trondheim, Norway: VLDB Endowment.

Gaizauskas, R., & Wilks, Y. (1998). Information Extraction: Beyond Document Retrieval. *Computational Linguistics and Chinese Language Processing*, *vol 3*, 17–60.

Geleijnse, G., Korst, J., & Pronk, V. (2006). Google-based Information Extraction. In *Proceedings of the 6th Dutch-Belgian Information Retrieval Workshop (DIR 2006)* (pp. 39–46). Delft, the Netherlands.

Godfrey, D., Johns, C., Meyer, C. D., Race, S., & Sadek, C. (2014). A Case Study in Text Mining: Interpreting Twitter Data From World Cup Tweets. *CoRR*, *abs/1408.5*.

Gómez-Pérez, A., Fernández-López, M., & Corcho-García, O. (2004). *Ontological Engineering*. Springer/Heidelberg.

Gruber, T. R. (1995). Toward principles for the design of ontologies used for knowledge sharing? *International Journal of Human-Computer Studies*, *43*(5–6), 907–928. doi:10.1006/ijhc.1995.1081

Hage, W. R. Van. (2005). A Method to Combine Linguistic Ontology-Mapping Techniques. In *4th International Semantic Web Conference (ISWC 2005)* (pp. 732–744). Galway, Ireland. doi:10.1.1.106.13

Handschuh, S., Staab, S., & Studer, R. (2003). Leveraging Metadata Creation for the Semantic Web with CREAM. In A. Günter, R. Kruse, & B. Neumann (Eds.), *26th Annual German Conference on AI, KI 2003: Advances in Artificial Intelligence SE - 2* (Vol. 2821, pp. 19–33). Hamburg, Germany: Springer Berlin Heidelberg. doi:10.1007/978-3-540-39451-8_2

Hanson, S. J., Cowan, J. D., & Giles, C. L. (1993). Advances in Neural Information Processing Systems 5. In *{[NIPS} Conference*. Denver, Colorado, USA: Morgan Kaufmann.

Hearst, M. A. (1992). Automatic Acquisition of Hyponyms from Large Text Corpora. In *In Proceedings of the 14th International Conference on Computational Linguistics* (pp. 539–545).

Hotho, A., Maedche, A., & Staab, S. (2002). Ontology-based Text Document Clustering. *KÜNSTLICHE INTELLIGENZ*, *4*, 48–54.

Ifrim, G., Shi, B., & Brigadir, I. (2014). Event Detection in Twitter using Aggressive Filtering and Hierarchical Tweet Clustering. In *SNOW-DC@WWW* (pp. 33–40).

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data Clustering: A Review. *ACM Comput. Surv.*, *31*(3), 264–323. doi:10.1145/331499.331504

Jansen, B. J., Zhang, M., Sobel, K., & Chowdury, A. (2009). Twitter Power: Tweets As Electronic Word of Mouth. *Journal of the American Society for Information Science and Technology*, *60*(11), 2169–2188. doi:10.1002/asi.v60:11

Jiang, J. J., & Conrath, D. W. (1997). Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. *CoRR, cmp-lg/970*.

Karkaletsis, V., Fragkou, P., Petasis, G., & Iosif, E. (2011). Ontology Based Information Extraction from Text. In *Multimedia Information Extraction, Lecture Notes on Artificial Intelligence 6050* (pp. 89–109). doi:10.1007/978-3-642-20795-2_4

Keller, F., Lapata, M., & Ourioupina, O. (2002). Using the Web to Overcome Data Sparseness. In *IN PROCEEDINGS OF EMNLP-02* (pp. 230–237).

Kiefer, C., Bernstein, A., & Locher, A. (2008). Adding Data Mining Support to SPARQL via Statistical Relational Learning Methods. In *Proceedings of the 5th European Semantic Web Conference on The Semantic Web: Research and Applications* (pp. 478–492). Berlin, Heidelberg: Springer-Verlag.

Kim, Y., & Shim, K. (2011). TWITOBI: A Recommendation System for Twitter Using Probabilistic Modeling. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on* (pp. 340–349). doi:10.1109/ICDM.2011.150

Kim, Y., & Shim, K. (2014). TWILITE: A recommendation system for Twitter using a probabilistic model based on latent Dirichlet allocation. *Information Systems*, *42*(0), 59–77. doi:http://dx.doi.org/10.1016/j.is.2013.11.003

King, B. (1967). Step-Wise Clustering Procedures. *Journal of the American Statistical Association*, *62*(317), 86–101. doi:10.1080/01621459.1967.10482890

Kiryakov, A., Popov, B., Terziev, I., Manov, D., & Ognyanoff, D. (2004). Semantic annotation, indexing, and retrieval. *Web Semantics Science Services and Agents on the World Wide Web*, *2*, 49–79. doi:10.1016/j.websem.2004.07.005

Kiyavitskaya, N., Zeni, N., Cordy, J. R., Mich, L., & Mylopoulos, J. (2005). Semi-automatic semantic annotations for web documents. In *2nd Italian Semantic Web Workshop on Semantic Web Applications and Perspectives, SWAP 2005 (CEUR Workshop Proceedings)* (Vol. 166, pp. 210–225). Trento, Italy.

Koivunen, M. (2005). 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 , CEUR Workshop Proceedings* (p. pp 5–17). Heraklion, Crete.

Kunneman, F., & van den Bosch, A. (2014). Event detection in Twitter: A machine-learning approach based on term pivoting. In and J. K. F. Grootjen, M. Otworowska (Ed.), *Proceedings of the 26th Benelux Conference on Artificial Intelligence* (pp. 65–72).

Kywe, S. M., Hoang, T.-A., Lim, E.-P., & Zhu, F. (2012). On recommending hashtags in twitter networks. In *Proceedings of the 4th international conference on Social Informatics* (pp. 337–350). Berlin, Heidelberg: Springer-Verlag. doi:10.1007/978-3-642-35386-4_25

Lamparter, S., Ehrig, M., & Tempich, C. (2004). Knowledge extraction from classification schemas. In *On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE* (pp. 618–636).

Lavanya, S., & Kavipriya, R. (2014). A Survey on Event Detection in News Streams. *International Journal of Computer Science Trends and Technology (IJCST), Volume 2*(Issue 5), 33–35.

Leacock, C., & Chodorow, M. (1998). Combining local context and WordNet similarity for word sense identification. In C. Fellfaum (Ed.), *MIT Press* (pp. 265–283). Cambridge, Massachusetts.

Lee, J. H., Kim, M. H., & Lee, Y. J. (1993). Information Retrieval Based on Conceptual Distance in IS-A Hierarchies. *Journal of Documentation*, *49*(2), 188–207.

Lee, W.-J., Oh, K.-J., Lim, C.-G., & Choi, H.-J. (2014). User profile extraction from Twitter for personalized news recommendation. In *Advanced Communication Technology (ICACT), 2014 16th International Conference on* (pp. 779–783). doi:10.1109/ICACT.2014.6779068

Legendre, P., & Legendre, L. F. J. (1998). *Numerical Ecology*. Elsevier Science.

Li, Y., Bandar, Z. A., & McLean, D. (2003). An Approach for Measuring Semantic Similarity Between Words Using Multiple Information Sources. *IEEE Trans. on Knowl. and Data Eng.*, *15*(4), 871–882. doi:10.1109/TKDE.2003.1209005

Li, Z., & Ramani, K. (2007). Ontology-based design information extraction and retrieval. *AI EDAM: Artificial Intelligence for Engineering Design, Analysis, and Manufacturing*, *21*(02), 137–154.

Lin, D. (1998). Automatic Retrieval and Clustering of Similar Words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 2* (pp. 768–774). Stroudsburg, PA, USA: Association for Computational Linguistics. doi:10.3115/980691.980696

Liu, C. L. (1968). *Introduction to combinatorial mathematics*. McGraw-Hill.

Lloyd, S. (1982). Least squares quantization in PCM. *Information Theory, IEEE Transactions on*, *28*(2), 129–137. doi:10.1109/TIT.1982.1056489

Ma, Z., Sun, A., Yuan, Q., & Cong, G. (2014). Tagging Your Tweets: A Probabilistic Modeling of Hashtag Annotation in Twitter. In *Proceedings of the 23rd ACM International Conference on Conference on Information and*

Knowledge Management (pp. 999–1008). New York, NY, USA: ACM. doi:10.1145/2661829.2661903

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*. Berkeley, Calif.: University of California Press.

Maedche, A., Neumann, G., & Staab, S. (2003). Bootstrapping an ontology-based information extraction system. In *Intelligent exploration of the web* (pp. 345–359).

Martínez, S., Valls, A., & Sánchez, D. (2012). Semantically-grounded construction of centroids for datasets with textual attributes. *Knowledge-Based Systems*, *35*(0), 160–172. doi:10.1016/j.knosys.2012.04.030

Matuszek, C., Witbrock, M. J., Kahlert, R. C., Cabral, J., Schneider, D., Shah, P., & Lenat, D. B. (2005). Searching for Common Sense: Populating Cyc from the Web. In *Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference* (pp. 1430–1435). Pittsburgh, Pennsylvania, USA.

Mausam, Schmitz, M., Bart, R., Soderland, S., & Etzioni, O. (2012). Open Language Learning for Information Extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (pp. 523–534). Stroudsburg, PA, USA: Association for Computational Linguistics.

Mazzia, A., & Juett, J. (2011). Suggesting hashtags on twitter. In *EECS 545 Project, Winter Term.*

McCallum, A. (2003). Efficiently Inducing Features of Conditional Random Fields. In *Proceedings of the Nineteenth Conference on Uncertainty in Artificial Intelligence* (pp. 403–410). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

McDowell, L. K., & Cafarella, M. (2008). Ontology-driven, Unsupervised Instance Population. *Web Semantics: Science, Services and Agents on the World Wide Web*, *6*(3), 218–236. doi:10.1016/j.websem.2008.04.002

Meij, E., Weerkamp, W., & de Rijke, M. (2012). Adding semantics to microblog posts. In *Proceedings of the fifth ACM international conference on Web search and data mining - WSDM '12* (p. 563). New York, New York, USA: ACM Press. doi:10.1145/2124295.2124364

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

Mikheev, A., & Finch, S. (1997). A Workbench for Finding Structure in Texts. In *Proceedings of the Fifth Conference on Applied Natural Language Processing* (pp. 372–379). Stroudsburg, PA, USA: Association for Computational Linguistics. doi:10.3115/974557.974611

Miller, G. A., & Charles, W. G. (1991). Contextual correlates of semantic similarity. *Language and Cognitive Processes*. doi:10.1080/01690969108406936

Morbach, J., Yang, A., & Marquardt, W. (2007). OntoCAPE—A large-scale ontology for chemical process engineering. *Engineering Applications of Artificial Intelligence*, *20*(2), 147–161. doi:10.1016/j.engappai.2006.06.010

Moreno, A., Isern, D., & Fuentes, A. C. L. (2013). Ontology-based information extraction of regulatory networks from scientific articles with case studies for Escherichia coli. *Expert Systems with Applications*, *40*(8), 3266–3281. doi:http://dx.doi.org/10.1016/j.eswa.2012.12.090

Moreno, A., Valls, A., Isern, D., Marin, L., & Borràs, J. (2013). Sigtur/e-destination: ontology-based personalized recommendation of tourism and leisure activities. *Engineering Applications of Artificial Intelligence*, *26*(1), 633–651.

Moreno, A., Valls, A., Martínez, S., Vicient, C., Marín, L., & Mata, F. (2014). Personalised recommendations based on novel semantic similarity and clustering procedures. *AI Communications*. doi:10.3233/AIC-140612

Moreno, A., Valls, A., Mata, F., Martínez, S., Marín, L., & Vicient, C. (2013). A semantic similarity measure for objects described with multi-valued categorical attributes. In *Frontiers in Artificial Intelligence and Applications* (Vol. 256, pp. 263–272). doi:10.3233/978-1-61499-320-9-263

Nakashole, N., Theobald, M., & Weikum, G. (2011). Scalable Knowledge Harvesting with High Precision and High Recall. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining* (pp. 227–236). New York, NY, USA: ACM. doi:10.1145/1935826.1935869

Neches, R., Fikes, R., Finin, T., Gruber, T., Patil, R., Senator, T., & Swartout, W. R. (1991). Enabling Technology for Knowledge Sharing. *AI Mag.*, *12*(3), 36–56.

Nédellec, C., & Nazarenko, A. (2005). Ontology and Information Extraction: a necessary symbiosis. In B. M. Paul Buitelaar, Philipp Cimiano (Ed.), *Ontology Design and Population* (pp. 155–170). IOS Press.

Niekrasz, J., & Gruenstein, A. (2006). NOMOS: A Semantic Web Software Framework for Annotation of Multimodal Corpora. In *Proceedings of the 5th International Conference on Language Resources and Evaluation LREC 06* (pp. 21–27). Genoa, Italy.

Özdikiş, Ö., Şenkul, P., & Oguztüzün, H. (2012). Semantic expansion of hashtags for enhanced event detection in Twitter. In *The First International Workshop on Online Social Systems (WOSS)*.

Panasyuk, A., Yu, E. S.-L., & Mehrotra, K. G. (2014). Controversial Topic Discovery on Members of Congress with Twitter. *Procedia Computer Science*, *36*(0), 160–167. doi:http://dx.doi.org/10.1016/j.procs.2014.09.073

Pasca, M. (2004). Acquisition of Categorized Named Entities for Web Search. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management* (pp. 137–145). New York, NY, USA: ACM. doi:10.1145/1031171.1031194

Petkos, G., Papadopoulos, S., Aiello, L., Skraba, R., & Kompatsiaris, Y. (2014). A soft frequent pattern mining approach for textual topic detection. In ACM (Ed.), *Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics (WIMS14)* (p. 25). Thessaloniki, Greece.

Petkos, G., Papadopoulos, S., & Kompatsiaris, Y. (2014). Two-level message clustering for topic detection in Twitter. In S. Papadopoulos, D. Corney, & L. M. Aiello (Eds.), *Proceedings of the SNOW 2014 Data Challenge co-located with 23rd International World Wide Web Conference (WWW 2014)* (Vol. 1150, pp. 49–56). Seoul, Korea: CEUR-WS.org.

Petrović, S., Osborne, M., & Lavrenko, V. (2010). Streaming First Story Detection with Application to Twitter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics* (pp. 181–189). Stroudsburg, PA, USA: Association for Computational Linguistics.

Phuvipadawat, S., & Murata, T. (2010). Breaking News Detection and Tracking in Twitter. In *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on* (Vol. 3, pp. 120–123). doi:10.1109/WI-IAT.2010.205

Pirró, G., & Seco, N. (2008). Design, Implementation and Evaluation of a New Semantic Similarity Metric Combining Features and Intrinsic Information Content. In *Proceedings of the OTM 2008 Confederated International Conferences, CoopIS, DOA, GADA, IS, and ODBASE 2008. Part II on On the Move to Meaningful Internet Systems* (pp. 1271–1288). Berlin, Heidelberg: Springer-Verlag. doi:10.1007/978-3-540-88873-4_25

Porter, M. F. (1997). Readings in Information Retrieval. In K. Sparck Jones & P. Willett (Eds.), (pp. 313–316). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Pöschko, J. (2011). Exploring Twitter Hashtags. *The Computing Research Repository (CoRR)*.

Quercia, D., Askham, H., & Crowcroft, J. (2012). TweetLDA: Supervised Topic Classification and Link Prediction in Twitter. In *Proceedings of the 4th Annual ACM Web Science Conference* (pp. 247–250). New York, NY, USA: ACM. doi:10.1145/2380718.2380750

Rada, R., Mili, H., Bicknell, E., & Blettner, M. (1989). Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man, and Cybernetics*, *19*(1), 17–30. doi:10.1109/21.24528

Rajani, N. F. N., McArdle, K., & Baldridge, J. (2014). Extracting Topics Based on Authors, Recipients and Content in Microblogs. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval* (pp. 1171–1174). New York, NY, USA: ACM. doi:10.1145/2600428.2609537

Ramage, D. ., Dumais, S. ., & Liebling, D. . (2010). Characterizing microblogs with topic models. In *ICWSM 2010 - Proceedings of the 4th International AAAI Conference on Weblogs and Social Media* (pp. 130–137).

Resnik, P. (2011). Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. *CoRR, abs/1105.5*.

Richardson, R., Smeaton, A. F., & Murphy, J. (1994). *Using WordNet as a Knowledge Base for Measuring Semantic Similarity between Words*.

Roberts, A., Gaizauskas, R., Hepple, M., Davis, N., Demetriou, G., Guo, Y., … Wheeldin, B. (2007). The CLEF corpus: semantic annotation of clinical text. In *AMIA 2007 Annual Symposium* (pp. 625–629). Chicago, USA: American Medical Informatics Association.

Romero, M., Moreo, A., Castro, J. L., & Zurita, J. M. (2012). Using Wikipedia concepts and frequency in language to extract key terms from support documents. *Expert Systems with Applications*, *39*(18), 13480–13491. doi:http://dx.doi.org/10.1016/j.eswa.2012.07.011

Rosa, H., Batista, F., & Carvalho, J. P. (2014). Twitter Topic Fuzzy Fingerprints. *Fuzzy Systems (FUZZ-IEEE), 2014 IEEE International Conference on*. doi:10.1109/FUZZ-IEEE.2014.6891781

Rosso, P., Montes Y Gómez, M., Buscaldi, D., Pancardo-Rodríguez, A., & Pineda, L. V. (2005). Two Web-Based Approaches for Noun Sense Disambiguation. In A. F. Gelbukh (Ed.), *CICLing* (Vol. 3406, pp. 267–279). Springer.

Rubenstein, H., & Goodenough, J. B. (1965). Contextual Correlates of Synonymy. *Commun. ACM*, *8*(10), 627–633. doi:10.1145/365628.365657

Russell, M. G., Flora, J., Strohmaier, M., Poschko, J., & Rubens, N. (2011). Semantic Analysis of Energy-Related Conversations in Social Media: A Twitter Case Study. In *International Conference of Persuasive Technology (Persuasive 2011)*. Columbus, OH, USA.

Sachidanandan, S. (2014). *SCAT: A system for concept annotation of tweets*. International Institute of Information Technology Hyderabad, India.

Sánchez, D., Batet, M., Isern, D., & Valls, A. (2012a). Ontology-based semantic similarity: A new feature-based approach. *Expert Systems with Applications*, *39*(9), 7718–7728. doi:10.1016/j.eswa.2012.01.082

Sánchez, D., Batet, M., Isern, D., & Valls, A. (2012b). Ontology-based semantic similarity: A new feature-based approach. *Expert Systems with Applications*, *39*(9), 7718–7728. doi:http://dx.doi.org/10.1016/j.eswa.2012.01.082

Sánchez, D., Isern, D., & Millan, M. (2011). Content annotation for the semantic web: An automatic web-based approach. *Knowledge and Information Systems*, *27*, 393–418. doi:10.1007/s10115-010-0302-3

Sanderson, M., & Croft, B. (1999). Deriving Concept Hierarchies from Text. In *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 206–213). New York, NY, USA: ACM. doi:10.1145/312624.312679

Sankaranarayanan, J., Samet, H., Teitler, B. E., Lieberman, M. D., & Sperling, J. (2009). TwitterStand: News in Tweets. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic*

Information Systems (pp. 42–51). New York, NY, USA: ACM. doi:10.1145/1653771.1653781

Schroeter, R., & Hunter, J. (2003). Vannotea - A Collaborative Video Indexing, Annotation and Discussion System for Broadband Networks. In *Knowledge Markup and Semantic Annotation Workshop, K-CAP 03* (pp. 9–26). Sanibel, Florida.

Shamma, D. A., Kennedy, L., & Churchill, E. F. (2011). Peaks and Persistence: Modeling the Shape of Microblog Conversations. In *Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work* (pp. 355–358). New York, NY, USA: ACM. doi:10.1145/1958824.1958878

Skounakis, M., Craven, M., & Ray, S. (2003). Hierarchical Hidden Markov Models for Information Extraction. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence* (pp. 427–433). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Sneath, P. H. A., & Sokal, R. R. (1973). *Numerical Taxonomy: The Principles and Practice of Numerical Classification*. London, UK: W. H. Freeman.

Soderland, S. (1999). Learning Information Extraction Rules for Semi-Structured and Free Text. *Machine Learning*, *34*(1-3), 233–272. doi:10.1023/A:1007562322031

Sørensen, T. (1948). A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons. *Biologiske Skrifter*, *5*, 1–34.

Spackman, K. A. (2004). SNOMED CT milestones: endorsements are added to already-impressive standards credentials. *Healthcare Informatics: The Business Magazine for Information and Communication Systems, 21*(9), 54–56.

Staab, S., & Maedche, A. (2000). Ontology Engineering beyond the Modeling of Concepts and Relations. In *Proceedings of the ECAI2000 Workshop on Ontologies and ProblemSolving Methods Berlin August 2122 2000* (pp. 8.1–8.8).

Stevenson, M., & Gaizauskas, R. (2000). Using Corpus-derived Name Lists for Named Entity Recognition. In *Proceedings of the sixth conference on Applied natural language processing* (pp. 290–295). doi:10.3115/974147.974187

Stilo, G., & Velardi, P. (2014). Time Makes Sense: Event Discovery in Twitter Using Temporal Similarity. In *IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2014* (Vol. 2, pp. 186–193). doi:10.1109/WI-IAT.2014.97

Stojanovski, D., Dimitrovski, I., & Madjarov, G. (2014). TweetViz : Twitter Data Visualization. In *Data Mining and Data Warehouses (SiKDD 2014)* (pp. 1–4). Ljubljana, Slovenia.

Studer, R., Benjamins, V. R., & Fensel, D. (1998). Knowledge Engineering: Principles and Methods. *Data Knowl. Eng.*, *25*(1-2), 161–197. doi:10.1016/S0169-023X(97)00056-6

Stumme, G., Ehrig, M., Handschuh, S., Hotho, A., Maedche, A., Motik, B., … Zacharias, V. (2003). *The {K}arlsruhe view on ontologies*.

Suchanek, F. M., Kasneci, G., & Weikum, G. (2008). YAGO: A Large Ontology from Wikipedia and WordNet. *Web Semantics*, *6*(3), 203–217.

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*. Doubleday.

Teufl, P., & Kraxberger, S. (2011). Extracting semantic knowledge from twitter. In *Proceedings of the Third IFIP WG 8.5 international conference on Electronic participation* (pp. 48–59). Berlin, Heidelberg: Springer-Verlag.

Tsur, O., Littman, A., & Rappoport, A. (2012). Scalable Multi Stage Clustering of Tagged Micro-messages. In *Proceedings of the 21st International Conference Companion on World Wide Web* (pp. 621–622). New York, NY, USA: ACM. doi:10.1145/2187980.2188157

Tsur, O., Littman, A., & Rappoport, A. (2013). Efficient Clustering of Short Messages into General Domains. In E. Kiciman, N. B. Ellison, B. Hogan, P. Resnick, & I. Soboroff (Eds.), *Proceedings of the 7th International AAAI Conference on Weblogs and Social Media (ICWSM 2013)* (pp. 621–630). The AAAI Press.

Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010). Predicting elections with Twitter: What 140 characters reveal about political sentiment. In *ICWSM 2010 - Proceedings of the 4th International AAAI Conference on Weblogs and Social Media* (pp. 178–185).

Turney, P. D. (2001). Mining the Web for Synonyms: PMI-IR Versus LSA on TOEFL. In *Proceedings of the 12th European Conference on Machine Learning* (pp. 491–502). London, UK, UK: Springer-Verlag.

Tversky, A. (1977). Features of similarity. *Psychological Review*, *84*, 327–352.

Twitter. (2014). About Twitter: Twitter usage. Retrieved from https://about.twitter.com/company

Uschold, M., & Gruninger, M. (1996). Ontologies: Principles, methods and applications. *Knowledge Engineering Review*, *11*, 93–136.

Van Rijsbergen, C. J., Robertson, S. E., & Porter, M. F. (1980). *New Models in Probabilistic Information Retrieval*. Computer Laboratory, University of Cambridge.

Velardi, P., Navigli, R., Cucchiarelli, A., & Neri, F. (2006). Evaluation of OntoLearn, a methodology for automatic population of domain ontologies. In P. Buitelaar, P. Cimiano, & B. Magnini (Eds.), *Ontology Learning from Text: Methods, Applications and Evaluation*. IOS Press.

Veltri, G. A. (2012). Microblogging and nanotweets: Nanotechnology on Twitter. *Public Understanding of Science*. doi:10.1177/0963662512463510

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

Vicient, C., & Moreno, A. (2013). *A Study on the Influence of Semantics on the Analysis of Micro-blog Tags in the Medical Domain*. (A. Cuzzocrea, C. Kittl, D. E. Simos, E. Weippl, & L. Xu, Eds.)*Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 8127, pp. 446–459). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-40511-2

Vicient, C., Sanchez, D., & Moreno, A. (2011a). A Methodology to Discover Semantic Features from Textual Resources. In *Sixth International Workshop on Semantic Media Adaptation and Personalization (SMAP 2011)* (pp. 39–44). doi:10.1109/SMAP.2011.13

Vicient, C., Sanchez, D., & Moreno, A. (2011b). Ontology-Based Feature Extraction. In *Workshop on 4th Natural Language Processing and Ontology Engineering (NLPOE 2011) in IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT 2011)* (Vol. 3, pp. 189–192). Lyon (France): Ieee. doi:10.1109/WI-IAT.2011.199

Vicient, C., Sánchez, D., & Moreno, A. (2012). An automatic approach for ontology-based feature extraction from heterogeneous textual resources. *Engineering Applications of Artificial Intelligence*. doi:10.1016/j.engappai.2012.08.002

Wang, X., Wei, F., Liu, X., Zhou, M., & Zhang, M. (2011). Topic sentiment analysis in twitter: a graph-based hashtag sentiment classification approach. In *CIKM'11* (pp. 1031–1040).

Wang, Y., Liu, J., Qu, J., Huang, Y., Chen, J., & Feng, X. (2014). Hashtag Graph based Topic Model for Tweet Mining. In *Proceedings of the IEEE International Conference on Data Mining, (ICDM-2014)*. Shenzhen, China.

Wimalasuriya, D. C., & Dou, D. (2010). Ontology-based Information Extraction: An Introduction and a Survey of Current Approaches. *Journal of Information Science*, *36*(3), 306–323. doi:10.1177/0165551509360123

Wu, F., & Weld, D. S. (2010). Open Information Extraction Using Wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (pp. 118–127). Stroudsburg, PA, USA: Association for Computational Linguistics.

Wu, Z., & Palmer, M. (1994). Verbs semantics and lexical selection. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics* (pp. 133–138). Stroudsburg, PA, USA: Association for Computational Linguistics. doi:10.3115/981732.981751

Xiao, L., Wissmann, D., Brown, M., & Jablonski, S. (2004). Information Extraction from the Web: System and Techniques. *Applied Intelligence*, *21*(2), 195–224. doi:10.1023/B:APIN.0000033637.51909.04

Yang, M.-C., & Rim, H.-C. (2014). Identifying interesting Twitter contents using topical analysis. *Expert Systems with Applications*, *41*(9), 4330–4336. doi:http://dx.doi.org/10.1016/j.eswa.2013.12.051

Yang, S.-H., Kolcz, A., Schlaikjer, A., & Gupta, P. (2014). Large-scale High-precision Topic Modeling on Twitter. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1907–1916). New York, NY, USA: ACM. doi:10.1145/2623330.2623336

Yang, X., Gao, R., Han, Z., & Sui, X. (2012). Ontology-Based Hazard Information Extraction from Chinese Food Complaint Documents. In *Proceedings of the Third International Conference on Advances in Swarm*

Intelligence - Volume Part II (pp. 155–163). Berlin, Heidelberg: Springer-Verlag. doi:10.1007/978-3-642-31020-1_19

Yangarber, R., & Grishman, R. (2000). Unsupervised discovery of scenario-level patterns for information extraction. In *18th International Conference on Computational Linguistics, COLING 2000* (pp. 940–946). Saarbrücken, Germany.

Yi, K. (2013). A semantic similarity approach for linking tweet messages to Library of Congress subject headings using linked resources: a pilot study. In *24th ASIS SIG/CR Classification Research Workshop (Advances in Classification Research Online 24 81)* (pp. 43–50).

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

Zavitsanos, E., Tsatsaronis, G., Varlamis, I., & Paliouras, G. (2010). Scalable Semantic Annotation of Text Using Lexical and Web Resources. In S. Konstantopoulos, S. Perantonis, V. Karkaletsis, C. Spyropoulos, & G. Vouros (Eds.), *Artificial Intelligence: Theories, Models and Applications SE - 32* (Vol. 6040, pp. 287–296). Springer Berlin Heidelberg. doi:10.1007/978-3-642-12842-4_32

Zhang, X., Chen, X., Chen, Y., Wang, S., Li, Z., & Xia, J. (2014). Event detection and popularity prediction in microblogging. *Neurocomputing*, (0), -. doi:http://dx.doi.org/10.1016/j.neucom.2014.08.045

Zhang, Z., Gentile, A. L., & Augenstein, I. (2014). Linked Data As Background Knowledge for Information Extraction on the Web. *SIGWEB Newsl.*, (Summer), 5:1–5:9. doi:10.1145/2641730.2641735

Zhao, Y., & Karypis, G. (2002). *Criterion Functions for Document Clustering: Experiments and Analysis*. Minnesota. doi:10.1.1.16.6872

Zhu, J., Nie, Z., Liu, X., Zhang, B., & Wen, J.-R. (2009). StatSnowball: A Statistical Approach to Extracting Entity Relationships. In *Proceedings of the 18th International Conference on World Wide Web* (pp. 101–110). New York, NY, USA: ACM. doi:10.1145/1526709.1526724