TIN2009-11005

DAMASK

Data-Mining Algorithms with Semantic Knowledge

PROYECTO DE INVESTIGACIÓN
PROGRAMA NACIONAL DE INVESTIGACIÓN FUNDAMENTAL,
PLAN NACIONAL DE I+D+i 2008-2011
ÁREA TEMÁTICA DE GESTIÓN: Tecnologías informáticas

Deliverable D2

Ontology-Based Feature Extraction

Authored by
Carlos Vicient, Universitat Rovira i Virgili
David Sánchez, Universitat Rovira i Virgili
Antonio Moreno, Universitat Rovira i Virgili

Document information

project name:	DAMASK	
Project reference:	TIN2009-11005	
type of document:	Deliverable	
file name:		
version:		
authored by:	C.Vicient	15/07/2011
	D. Sánchez	
	A. Moreno	
co-authored by		
released by:		. .200
approved by:	Co-ordinator	A. Moreno

Document history

version	date	reason of modification
1.0	18.July.2011	A preliminary release of the manuscript includes the proposed methodology and the evaluation of the results.
2.0	25.July.2011	A final release of the manuscript includes the introduction and state of preavious documents, the proposed methodology, the evaluation of the the results and some conclutions.
	. .200	
	. .200	
	. .200	
	. .200	
	. .200	
	. .200	

Table of Contents

1 Introduction

The first task of DAMASK, called T1 - *Semantic integration of the information available in heterogeneous Web resources*, includes two preliminary subtasks, tasks 1.1 and 1.2 (see Figure 1). The former task discusses all related works in information extraction from the Web distinguishing algorithms to extract structured, semi-structured and non-structured resources. Deliverable D1 was the result of the task 1.1., and its aim was to make a state-of-the-art on Information Extraction techniques applied to Web resources. An internal report, which was the output of task 1.2 lists the types of data that can be extracted using the previous algorithms. As stated in that document, DAMASK takes into account three data types: Measurement, Nominal and Semantic.

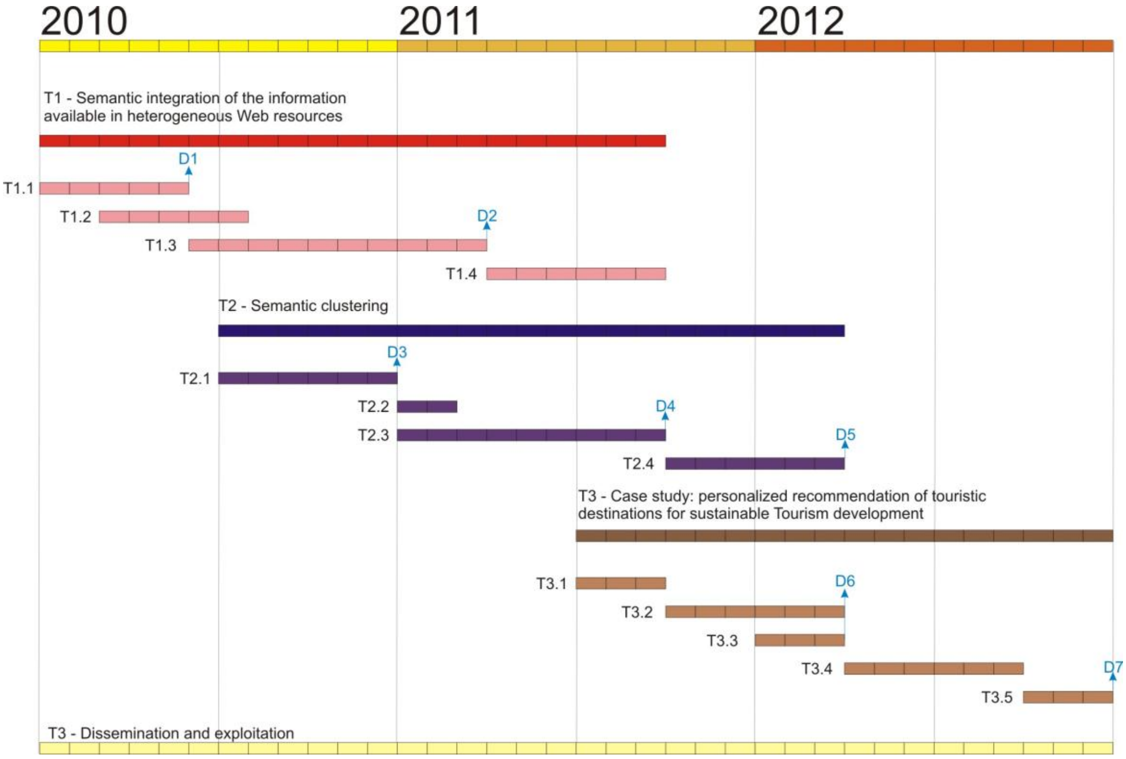


Figure 1: Tasks of DAMASK

This document (Deliverable D2) is the result of task 1.3. (see Figure 1), and its aim is to present the proposed methodology for Ontology-based feature extraction. The main goal of this task was to design and implement a novel method that was able to extract relevant features from a range of textual documents going from plain textual data to semi-structured resources. The designed methodology is able to take profit from pre-processed input when it is available in order to complement its own learning algorithms. The key point of the work is to complement the syntactical parsing and other natural language processing techniques with the knowledge contained in an input ontology (which ideally, should model the knowledge domain in which the posterior data analysis will be focused –e.g. touristic points of interest) in order to be able to:

1) identify relevant features describing a particular entity from textual data,

2) to associate, if applicable, extracted features to concepts contained in the input ontology. In this manner, the output of the system would consist on tagged features which can be directly exploited by semantically grounded data mining algorithms (e.g. clustering) in order to classify them.

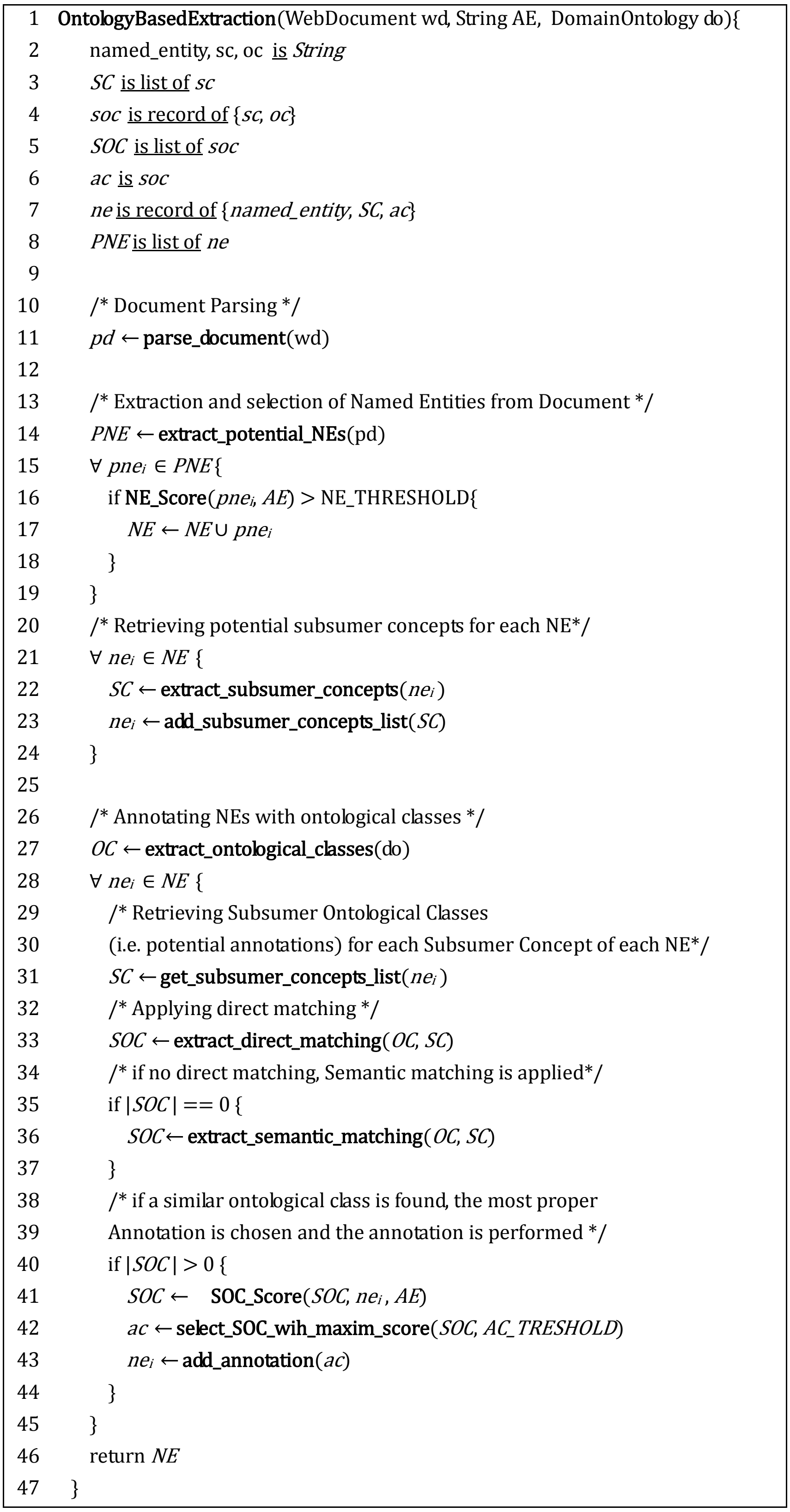
Finally, this approach has only taken into account semantic data types because the rest of data types were studied in other works like in [1], where Wikipedia Infoboxes were analysed in order to retrieve numerical attributes (i.e. measurement data types) and Nominal attributes from plain text.

2 Methodology

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

- In §2.1 the generic algorithm is described. It takes as input a Web-document to be analysed, a String that represents the analysed entity and a domain ontology which specifies the important concepts that should be extracted, and it returns, as a result, the relevant features (i.e. Named Entities) annotated semantically with concepts that appear in the input domain ontology.
- In §2.2 the applicability of the algorithm is studied in different kinds of resources. Specifically, plain text documents and Wikipedia articles have been taken into account.

2.1 Generic algorithm description



```
OntologyBasedExtraction(WebDocument wd, String AE,  DomainOntology do){
    named_entity, sc, oc  is String
    SC is list of sc
    soc is record of {sc, oc}
    SOC is list of soc
    ac is soc
    ne is record of {named_entity, SC, ac}
    PNE is list of ne

    /* Document Parsing */
    pd ← parse_document(wd)

    /* Extraction and selection of Named Entities from Document */
    PNE ← extract_potential_NEs(pd)
    ∀ $pne_i$ ∈ PNE{
      if NE_Score($pne_i$, AE) > NE_THRESHOLD{
        NE ← NE ∪ $pne_i$
      }
    }
    /* Retrieving potential subsumer concepts for each NE*/
    ∀ $ne_i$ ∈ NE {
      SC ← extract_subsumer_concepts($ne_i$)
      $ne_i$ ← add_subsumer_concepts_list(SC)
    }

    /* Annotating NEs with ontological classes */
    OC ← extract_ontological_classes(do)
    ∀ $ne_i$ ∈ NE {
      /* Retrieving Subsumer Ontological Classes
      (i.e. potential annotations) for each Subsumer Concept of each NE*/
      SC ← get_subsumer_concepts_list($ne_i$)
      /* Applying direct matching */
      SOC ← extract_direct_matching(OC, SC)
      /* if no direct matching, Semantic matching is applied*/
      if |SOC| == 0 {
        SOC ← extract_semantic_matching(OC, SC)
      }
      /* if a similar ontological class is found, the most proper
      Annotation is chosen and the annotation is performed */
      if |SOC| > 0 {
        SOC ←   SOC_Score(SOC, $ne_i$, AE)
        ac ← select_SOC_wih_maxim_score(SOC, AC_TRESHOLD)
        $ne_i$ ← add_annotation(ac)
      }
    }
    return NE
  }
```

Algorithm 1 Generic algorithm for the implemented methodology

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

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

The main steps will be explained in detail in the next subsections. In section 2.2 it is discussed how to take profit of two different types of resources (overwriting the aforementioned functions), namely plain text documents and semi-structured Wikipedia articles.

2.1.1 Document parsing

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

2.1.2 Named Entities

This step consists in extracting relevant named entities from the analysed document. NEs represent real world entities. In other words, named entities can be considered as instances of ontological concepts [2] (e.g. Tarragona is an instance of a city).

The function extract_potential_NEs (line 14) returns a set of Named Entities (PNE) but only a subset of the elements of PNE describes the main features of AE; the rest of the elements of PNE introduce noise because they are not directly related to the analysed entity (they just happen to appear in the Web page describing the entity but are not part of its basic distinguishing characteristics). Thus, it is necessary to have a way of separating the relevant NEs from the irrelevant ones (NE filtering, line 16). To do that, we use a Web-based co-occurrence measure that tries to assess the degree of relationship between AE and each NE. In fact, it has been stated that the amount and heterogeneity of information in the Web is so high that it can be assumed to approximate the real distribution of information in the world [3]. Concretely, a version of the Pointwise Mutual Information (PMI) relatedness measure adapted to the Web is computed [4].

$$NE_{score}(PNE_i, AE) = \frac{hits(PNE_i \ \& \ AE)}{hits(PNE_i)} \tag{1}$$

In the NEscore (Equation 1), concept probabilities are approximated by Web hit counts provided by a Web search engine. Finally, the NEs that have a score exceeding an empirically determined threshold (NE_THRESHOLD, line 16) are considered as relevant, whereas the rest are removed. The value of the threshold will determine a compromise between the precision and the recall of the system.

2.1.3 Semantic Annotation

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

In this area, some approaches have been proposed. One way to assess the relationship between two terms (which, in our case, would be a NE and an ontology class) is to use a general thesaurus like WordNet to compute a similarity measure based on the number of semantic links among the queried terms [5, 6]. However, those measures are hampered by WordNet's limited coverage of NEs and, in consequence; it is usually not possible to compute the similarity between a NE and an ontological class in this way.

There are approaches which try to discover automatically taxonomic relationships [7, 8], but they require a considerable amount of background documents and linguistic parsing.

Finally, another possibility is to compute the co-occurrence between each NE and each ontological class using Web-scale statistics as the relatedness measure [9], but this solution is not scalable because of the huge amount of required queries [10].

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

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

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

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

2.1.3.1 Discovering potential subsumer concepts

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

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

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

2.1.3.2 Ontology matching

We distinguish between two types of matching: Direct Matching and Semantic Matching. Moreover, there are situations in which there is evidence that a certain NE is related to several ontological classes. In this case, Web-based statistical measures are applied again in order to choose the most representative one (Class Selection, 2.1.3.2.3). These three steps are explained in the following subsections.

2.1.3.2.1 Direct Matching

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

2.1.3.2.2 Semantic Matching

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

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

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

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

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

Thus, the first step is, for each element of SC_i of each NE_i, look it up in WordNet. If it only has one definition (synset), the new subsumer candidates (synonyms, hypernyms and hyponyms) are retrieved. Otherwise, if the element of SC_i has more than one synset, it is necessary to choose the most suitable one (word sense disambiguation).

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

Next, an example of the method is explained. Table 1 depicts the input data of the problem. First three rows are the analysed entity, the named entity and its subsumer concept. The rest of the data indicates the performed query in order to retrieve web snippets and all the WordNet synsets for the subsumer concept.

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

Table 1 Semantic disambiguation example (part 1)

Table 2 shows a subset of the retrieved snippets, which represent the new context, and the final score when applying cosine distance between the context and the synset. As a final result, synset 1 obtains the highest score and synonyms, hyponyms and hypernyms are extracted from it. In this example, the related terms for the subsumer concepts retrieved from WordNet are: minster, church and church building.

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

Table 2 Semantic disambiguation example (part 2)

2.1.3.2.3 Class Selection

When more than one ontology class has been proposed (line 40) as annotation for a certain NE_i, the final step is to choose the most appropriate one. The selection is based on the relatedness between the Named Entity and each element of SOC_i, assessed again with the Web-based version of PMI. However, it must be noted that the elements of SOC_i can also be polysemous, and can be referring to different concepts depending on the context (line 41). So, in Eq.(2), the analysed entity AE has been introduced to contextualize the relationship of each element of SOC_i with NE_i.

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

The score (E.q.(2)) computes the probability of the co-occurrence of the named entity NE_i and each ontology class proposed for annotation SOC_{ij} from the Web hit count provided by a search engine when querying these two terms (contextualized with AE). Finally, only the annotation with the highest score which reaches the AC_Treshold (line 42) is annotated

2.2 Applying the algorithm to different types of Web resources

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

2.2.1 Extraction from raw text

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

2.2.1.1 Named Entities detection

The main problem related with NE detection is the fact that they are unstructured and unlimited by nature as is stated in [11]. This implies that, in most cases, these NEs are not contained in classical repositories as WordNet due to its potential size and its evolvability.

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

Supervised approaches try to detect NEs relying on a specific set of extraction rules learned from pre-tagged examples [12, 13], or predefined knowledge bases such as lexicons and gazetteers [14]. However, the amount of effort required to assemble large tagged sets or lexicons binds the NE recognition to either a limited domain (e.g., medical imaging), or a small set of predefined, broad categories of interest (e.g., persons, countries, organizations, products). This introduces compromises in the recall [15].

In unsupervised approaches like [16], it has been proposed to use a thesaurus as background knowledge (i.e., if a word does not appear in a dictionary, it is considered as a NE). Despite the fact that this approach is not limited by the size of the thesaurus, misspelled words are wrongly considered as NEs whereas correct NEs composed by a set of common words are rejected, providing inaccurate results.

Other approaches take into consideration the way in which NEs are presented in the specific language. Concretely, languages such as English distinguish proper names from other nouns through capitalization. The main problem is that basing the detection of NEs on individual observations may produce inaccurate results if no additional analyses are applied. For example, a noun phrase may be arbitrary capitalised to stress its importance or due to its placement within the text. However, this simple idea, combined with linguistic pattern analysis, as it has been applied by several authors [15, 17-19], provides good results without depending on manually annotated examples or specific categories.

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

First, the four modules of the OPENNLP parser (Sentence Detector, Tokenizer, Tagging and Chunking) are applied in order to analyse syntactically the input text of the Web document. The last module is able to tag Proper Nouns, which represent NEs, using an internal database, but this approach produces a low recall because of the reasons stated in section 2.1.2. For example, in [NP The/VB gothic/JJ cathedral/NN][VP of/VB][NP Barcelona/NNP], the noun phrase (NP) Barcelona is tagged as proper noun (/NNP) but, in [NP Tarragona/EX][VP is/NNS][NP a/JJS city/NN], NP Tarragona is erroneously not considered as proper noun. To avoid a supervised methodology based on a database, the output of OPENNLP has been complemented by capitalization heuristics where all Noun Phrases which contain one, or more than one, word begins with a capital letter has been considered as a NE and consequently a set of potential Named Entities (PNE) is detected. Thus, that all Noun Phrases which contain at least one word that begins with a capital letter are considered as a potential NE.

Table 3 shows an example of the extracted NE from the first fragment of text of Wikipedia article about Tarragona. Notice that only Spain is detected as a proper noun using only the natural language parser but applying capitalization heuristics the rest of Named Entities has been extracted.

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

Table 3 Set of extracted NE from Tarragona Wikipedia introduction

2.2.1.2 Discovering potential subsumer concepts

We use the standard Hearst's taxonomic linguistic patterns, which have proved their effectiveness to retrieve hyponym/hypernim relationships [20]. We exploit the Web as the corpus from which to extract the semantic evidences of the appearances of the patterns [21]. The main reason of using the Web as the corpus is because of the fact that explicit linguistic patterns are difficult to find in reduced corpora, that normally offer a relatively high precision but suffer from low recall.

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

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

Table 4 Patterns used to retrieve potential subsumer concepts

Table 4 summarizes the linguistic patterns that have been used (CONCEPT represents the retrieved potential subsumer concept and NE the Named Entity that is being studied).

2.2.2 Extraction from semi-structured Wikipedia documents

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

2.2.2.1 Named Entities detection

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

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

In order to illustrate these problems, the following fragment of text extracted from Wikipedia will be examined. "Barcelona is the capital and the most populous city of Catalonia and the second largest city in Spain, after Madrid, with a population of 1,621,537 within its administrative limits on a land area of 101.4 km2". In this text, there are four terms linked with other Wikipedia articles by means of internal links. Three of them are NE (Catalonia, Spain and Madrid) and they represent instances of things, the other one is a common noun which represents a concept (capital). The first wikilink is not a NE because the first part of the sentence is defining what the NE Barcelona is, and the person who edited the article considered that the term "capital" (which represents a concept) was important for a correct understanding of the text. Finally, the NE Madrid is bringing information of general purpose that is not directly related with Barcelona and, in consequence, it is not a relevant feature for describing the entity Barcelona.

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

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

Table 5 Subset of extracted NE from Barcelona Wikipedia article

Table 5 shows the first NE detected when using wikilinks. It is important to observe that this step is only concerning with detection and this NE will be filtered in next algorithm step. Notice that in "Barcelona Cathedral" and "Barcelona Pavilion" both cathedral and pavilion are common nouns but they are preceded by Barcelona specifying that it is referring a concrete real entity (i.e., a named entity).

2.2.2.2 Discovering potential subsumer concepts

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

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

Another limitation of Wikipedia categories is the fact that they do not always contain enough concepts to perform the matching among them and ontological classes. For instance, the NE Plaça de Catalunya is a square situated in the city centre of Barcelona. Its categories are Plazas in Barcelona, the Eixample and Central business districts but our ontology is focused on tourism domain and none of these concepts appears in it. By contrast, Plaça de Catalunya is considered as a tourist attraction in Barcelona and the concept visitor attraction is represented by the ontology. Fortunately, as mentioned before, Wikipedia categories are included in higher categories. Following the same example, Plazas in Barcelona belongs to the higher categories Squares and plazas by city, Geography of Barcelona and Visitor attractions in Barcelona. Notice that the last one represents the same concept that we want to find and in consequence a new potential subsumer concept has been found. Observe that higher levels of categories represent higher concepts and going up through categories the meaning of the NE which they represent could be lost. Moreover, the Wikipedia categorisation has been performed by hand and its structure is approximated by a directed acyclic graph fact which implies that it is not always possible to navigate through categories in a taxonomical way. For that reason, only two levels of categories have been used in our approach.

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

Table 6 exemplifies the subsumer concepts extractions from Wikipedia categories. This is only a temporary list of potential subsumer concepts but they will be selected as potential subsumer concepts when applying ontology matching.

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

Table 6 Subset of extracted potential subsumer for Barcelona NEs

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

2.2.3 Computational cost

The computational cost of the proposed method depends on the number of queries performed because they are the most expensive task [10]. We can distinguish five different tasks in which queries are performed: NE detection, NE filtering, subsumer concepts extraction, semantic disambiguation and class selection.

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

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

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

2.3 Conclusions

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

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

3 Evaluation

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

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

The recall is calculated by dividing the number of correct annotations performed by the system by the total of annotations the system should have annotated according to the expert's opinion (Equation (3)).

$$RECALL = \frac{\#Good_{annotations}}{\#Good_{annotations} + \#Unretrieved\_Good_{annotations}} \quad (3)$$

The Precision is the number of correct annotations according to the expert's opinion divided by the total number of annotations (Equation (4)).

$$PRECISIONS = \frac{\#Good_{annotations}}{\#Good_{annotations} + \#Bad_{annotations}} \quad (4)$$

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

- In §3.1, two ontologies used to test the system are presented.
- The influence of the thresholds used in the algorithm is studied for the city of Barcelona using one ontology, in §3.2.
- §3.3 evaluates the behaviour of our methodology using different domain ontologies.
- §3.4 shows a comparison between the analysis of a plain text versus that of a semi-structured Wikipedia document.
- Finally, in §3.5, some conclusions are extracted and the main advantages and drawbacks of each method are discussed.

3.1 Used ontologies

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

TourismOWL.owl ontology

This ontology models touristic points of interest for different kinds of tourist profiles. It was designed in a final year project [1] based on information extracted through Wikipedia articles. It consists of 315 classes and a depth of 5 hierarchical levels. Its main classes represent concepts related with administrative divisions (borough, city, country, village, etc.), buildings (commercial buildings, cultural buildings, religious buildings, sport buildings, etc.), festivals (art festivals, music festivals, carnival, etc.), landmarks (commemorate landmarks, geographical landmarks, memorial landmarks, etc.), museums (archaeology museum, history museum, science museum, etc.) and sports (football, basketball, hockey, formula one, etc.). See Annex 1.

Space.owl ontology

This ontology was found by looking up ontologies using the Web search engine SWOOGLE that is specialized in ontologies. The Space.owl ontology consists of 188 classes and a depth of 6 hierarchical levels. It contains concepts related with three main topics: geographical features (i.e., archipelago, beach, river, forest, etc.), geopolitical entities (i.e., country, capital, city, district, street, etc.) and places, which includes business places (factory, convention centre, etc.), private places (residential structure, home, etc.) and public places (educational and medical structures, entertainment places, shopping facilities, transportation connections, etc.). See Annex 2.

3.2 Influence of thresholds

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

Figure 2 shows a comparison between both thresholds. In the left column, the comparison is applied for Barcelona taking as input the plain text of the article, while the right column presents the results when taking profit of Wikipedia semi-structure.

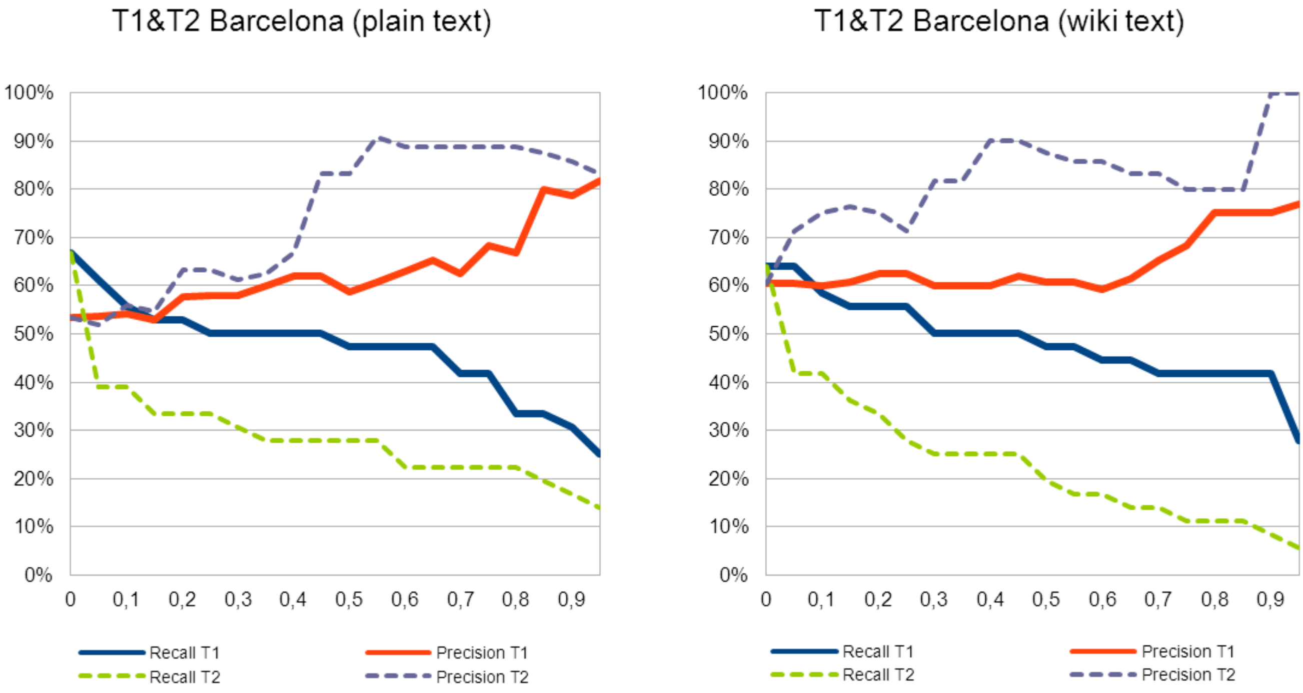


Figure 2 Influence of T1 and T2

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

3.3 Plain text vs. Wikipedia document

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

Figure 3 shows a comparison between the two methods (plain text and semi-structured) when applied to the cities of Barcelona and Canterbury. In the left column, the influence of the threshold for filtering NE (T1) and the threshold for selection annotation (T2) are studied for Barcelona, while the right column depicts the analysis for Canterbury.

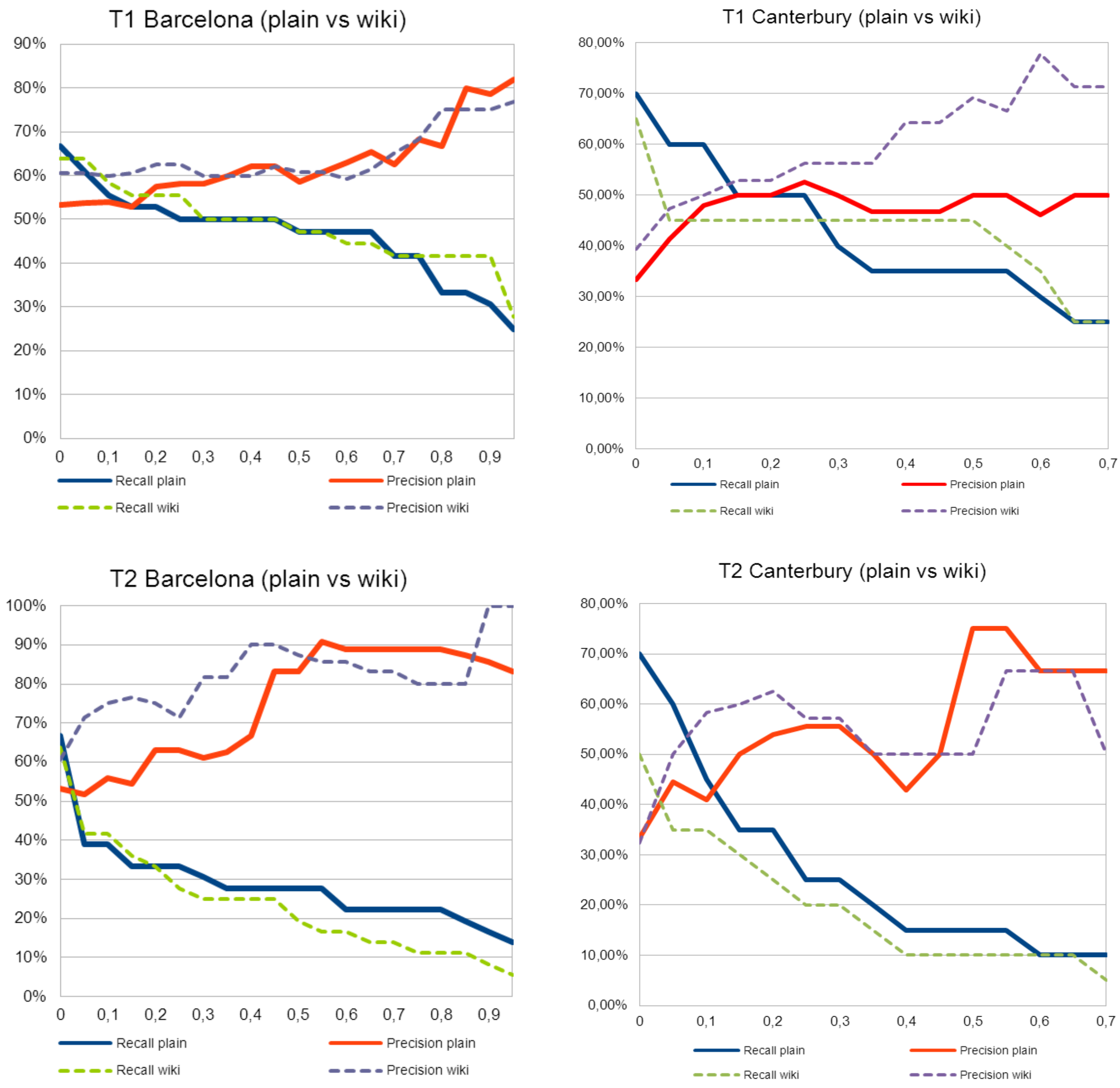


Figure 3 Plain text vs Wikipedia documents

The results show that the method is able to extract, in both cases, more than a 50% of the features marked for the domain expert. We also observe that the precision tend to keep or to improve when taking into account the semi-structure of the Wikipedia articles while the recall decreases. This is because in the first approach the whole textual content is analysed. This implies that there are more possibilities to detect representative features whereas the precision may be lower because there is a higher amount of unrepresentative features which add noise to the final results. On the opposite, using the second approach the set of analysed entities is limited to those manually annotated but, in contrast, the precision is higher because the potential candidates for each feature are extracted from Wikipedia categories (tagged and selected manually by a big community of users). It is important to note that, in any case, the analysis of Wikipedia articles is, as discussed in section 2.2.3, considerably faster than text, as the degree of analysis required to extract and annotate entities is reduced.

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

3.4 Influence of domain ontologies

This section compares the performance of our method using different domain ontologies. In both cases, the same Wikipedia article for the city of Barcelona has been tested using the ontologies stated above in section 3.1.

Figure 4 shows the results of the evaluation. All the graphs compare the recall and precision reached for the algorithm when applying it with different input ontologies (space.owl and tourismOWL.owl). The left column depicts the results when analysing the Wikipedia article as plain text while the right column represents the semi-structured approach. Although the objective of this evaluation is not to study the influence of thresholds, both analyses are represented based on the values of T1 and T2. This is because, in this way, it is possible to observe the global trend of analyses instead of fixing two values for both thresholds.

If we observe the left column, we can see that for both thresholds the precision obtained with the tourism ontology is higher. However, this fact does not happen, in the right column where the precision and recall for both ontologies is similar. This is because the tourism ontology was created based on the text of different Wikipedia articles about cities focused on touristic activities. This means that, for the first column, there are more direct matches than for the second one. Even though the difference between precision and recall thresholds is not high when using both ontologies, we can see that the results are a bit better for tourismOWL.owl ontology. These results show the importance of using a domain ontology proper to model the key concepts about the domain which is being studied in order to maximize the quality of the extracted features.

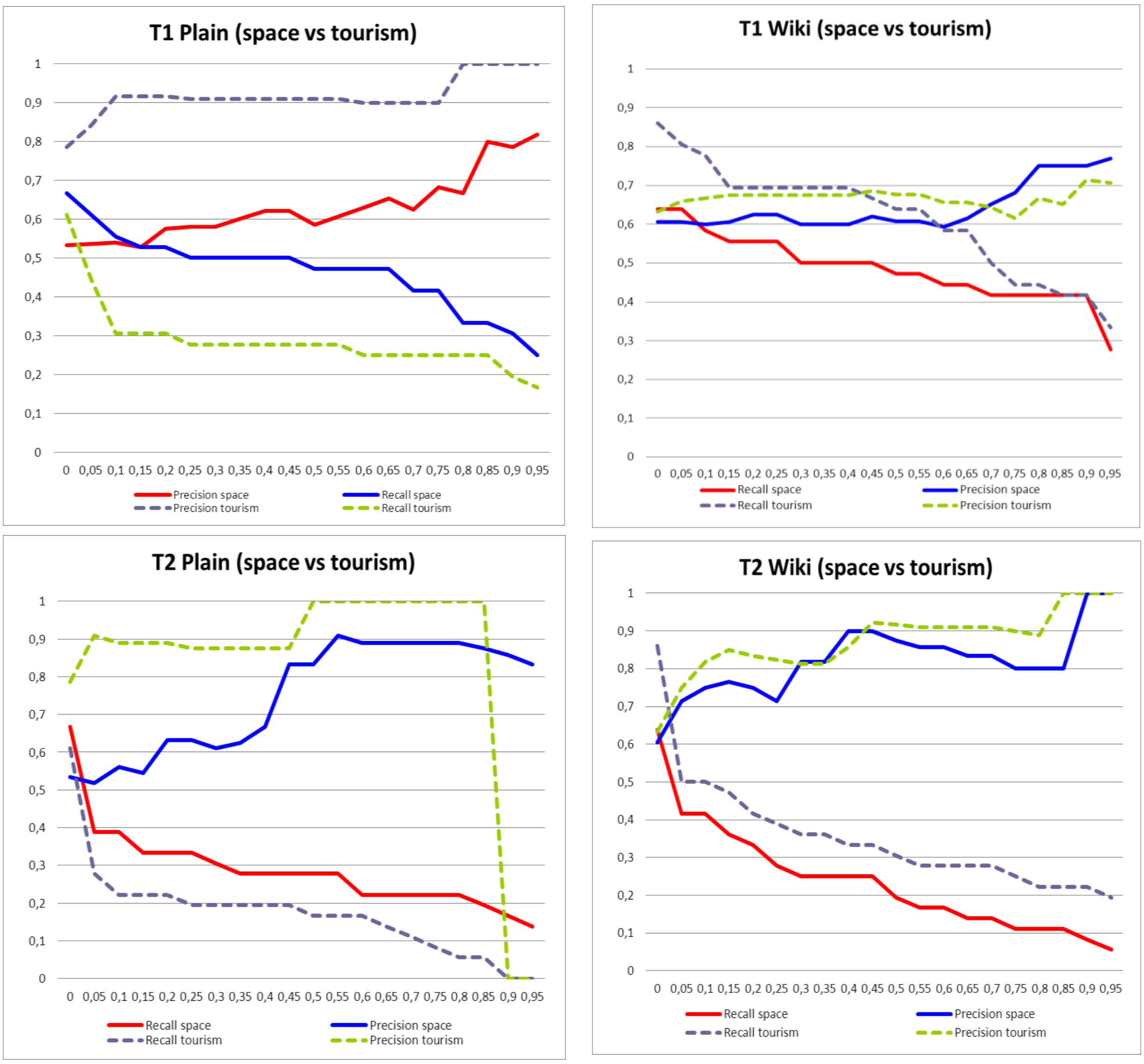


Figure 4 Influence of domain ontologies

3.5 Conclusions

In this section, an evaluation of the more relevant aspects of the feature extraction algorithm has been presented. The evaluation on any unsupervised automatic domain-independent extraction process is a hard task. On one hand, the evaluation has been performed through the intervention of a human expert in a particular domain that is represented by the input domain ontology. On the other hand, the final feature extraction and annotation is slanted by the precision and recall of each method used during the whole process (i.e., the natural language parser, the named entity detection heuristics, the inaccuracy of Web statistics and the relatedness measures). Considering these restrictions we can conclude that the final feature annotations are certainly usable, even reaching a 100% precision in some cases.

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

4 Summary

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

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

5 References

[1] C. Vicient, "Extracció basada en ontologies d'informació de destinacions turístiques a partir de la Wikipedia," Universitat Rovira i Virgili, Tarragona2009.

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

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

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

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

[6] Z. Wu and M. Palmer, "Verb semantics and lexical selection," in *32nd annual Meeting of the Association for Computational Linguistics*, 1994, pp. 133 -138.

[7] M. Sanderson and B. Croft, "Deriving concept hierarchies from text," Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, 1999.

[8] G. Bisson, C. Nédellec, and D. Cañamero, "Designing clustering methods for ontology building: The Mo'K workbench," Proceedings of the First Workshop on Ontology Learning OL'2000, Berlin, Germany, August 25, 2000, 2000.

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

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

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

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

[13] M. Stevenson and R. Gaizauskas, "Using corpus-derived name lists for named entity recognition," Proceedings of the sixth conference on Applied natural language processing, 2000.

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

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

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

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

[18] U. Hahn and K. Schnattinger, "Towards text knowledge engineering," Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence, 1998.

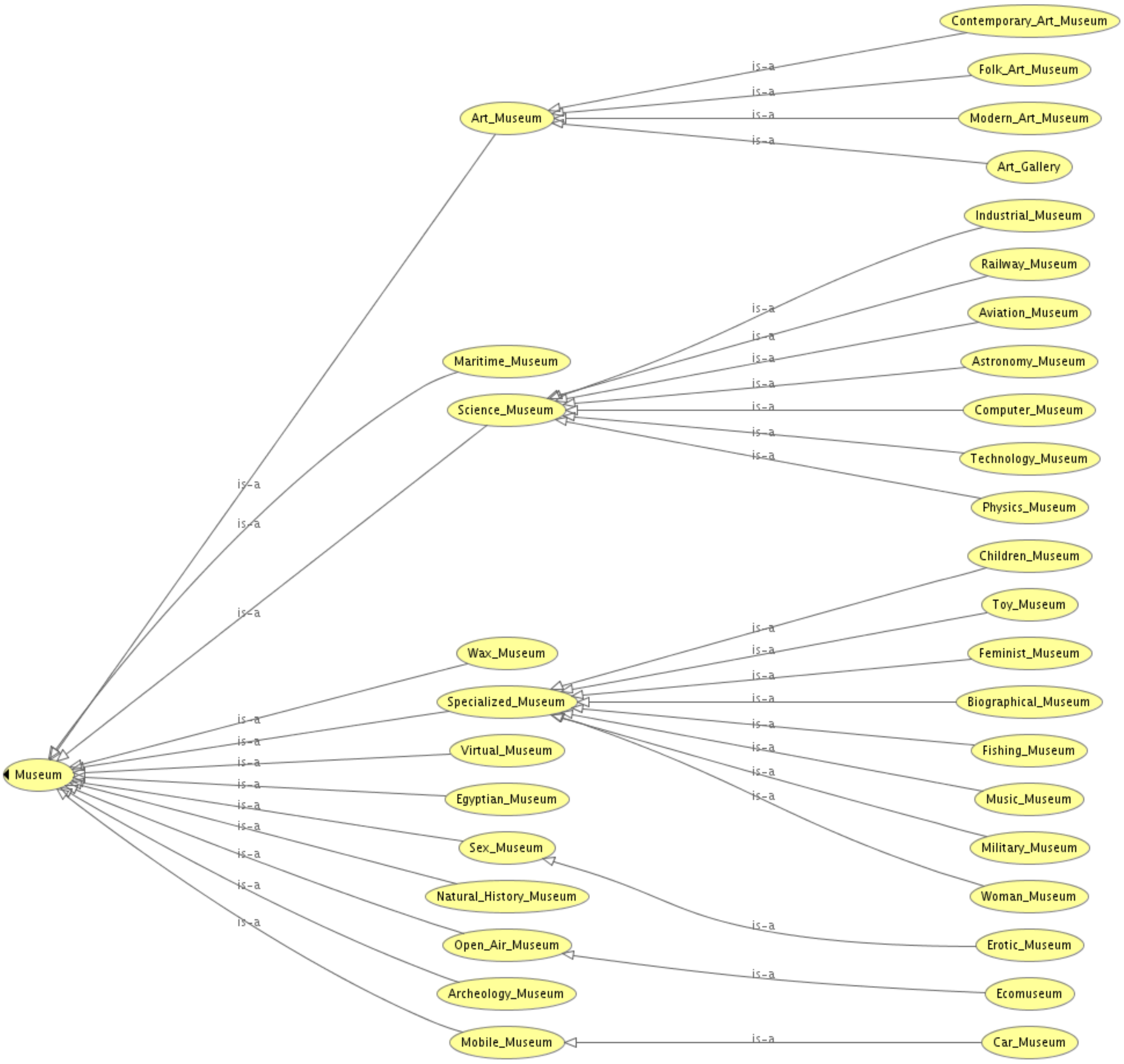
[19] D. Downey, M. Broadhead, and O. Etzioni, "Locating complex named entities in Web text," in *20th International Joint Conference on Artificial Intelligence, IJCAI 2007*, 2007, pp. 2733-2739.

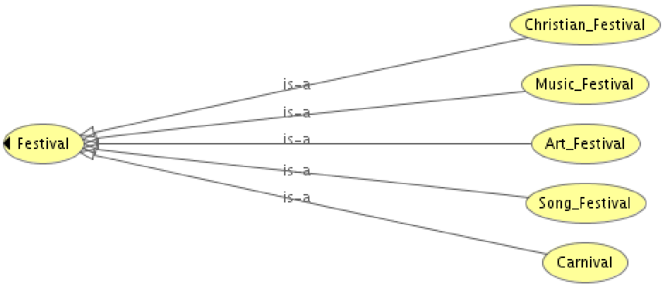
[20] M. A. Hearst, "Automatic acquisition of hyponyms from large text corpora," in *14th conference on Computational linguistics - Volume 2, COLING 92*, 1992, pp. 539 - 545.

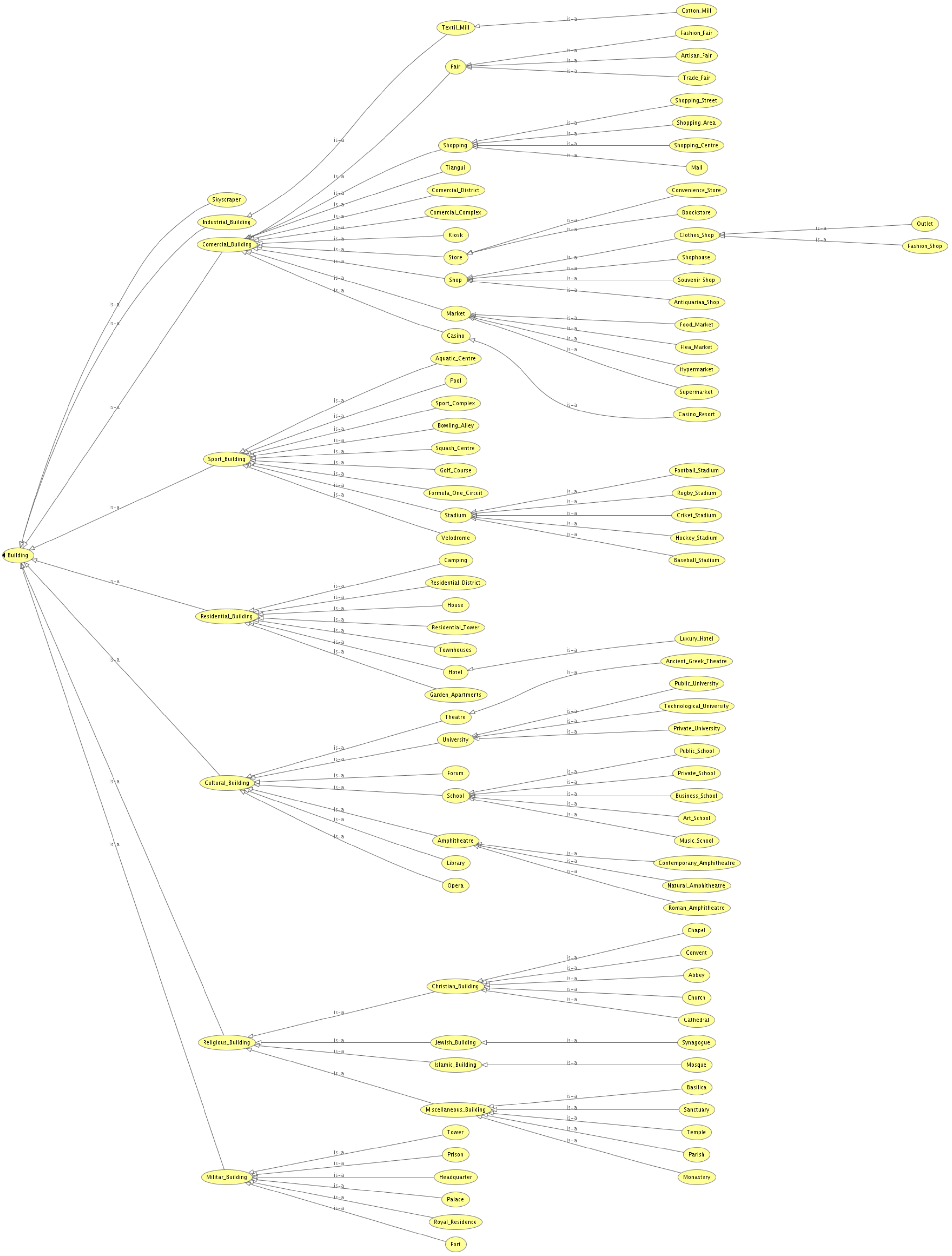
[21] B. Rozenfeld and R. Feldman, "Self-supervised relation extraction from the Web," *Knowl. Inf. Syst.*, vol. 17, pp. 17-33, 2008.

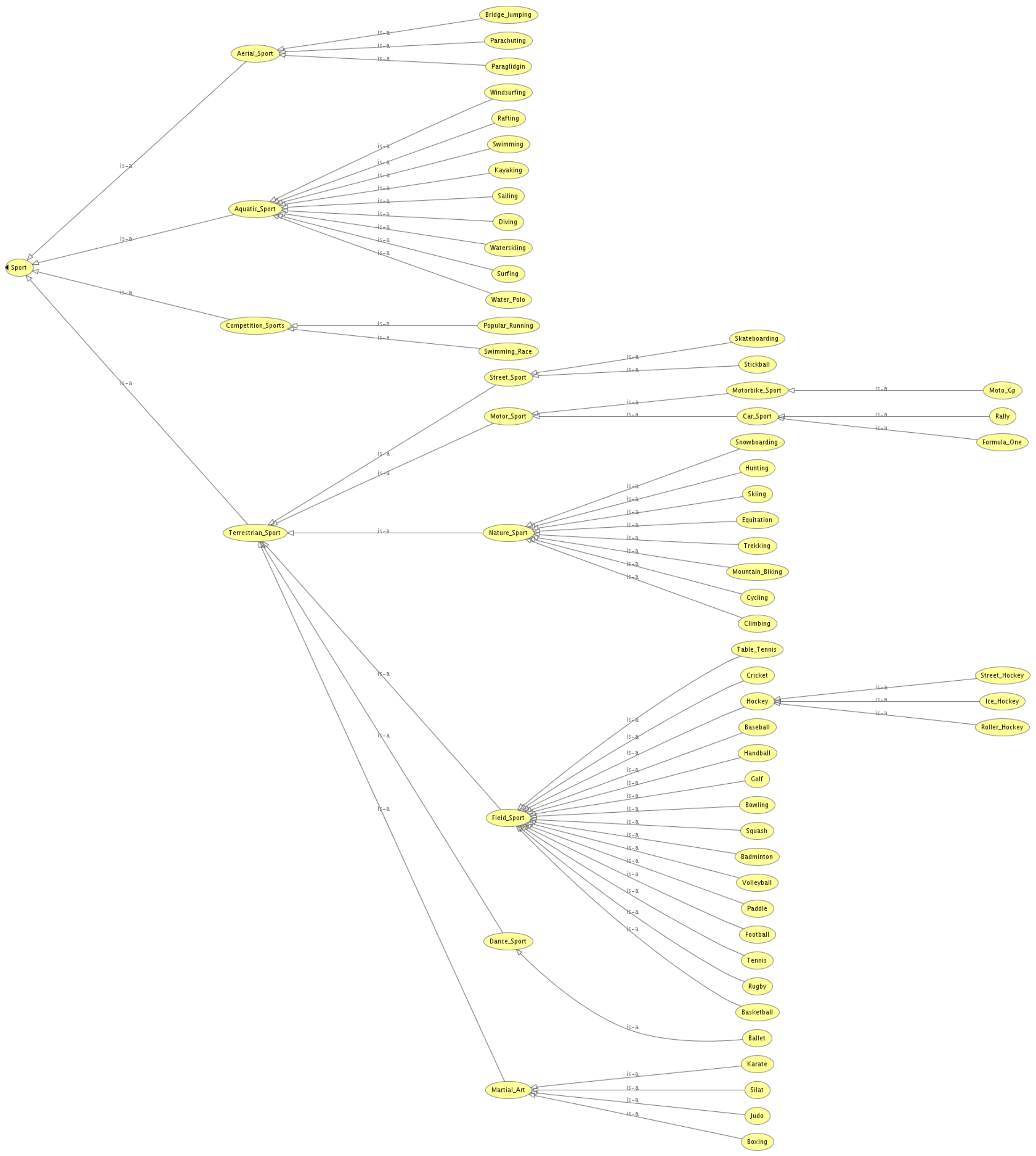
Annex I – TourismOWL

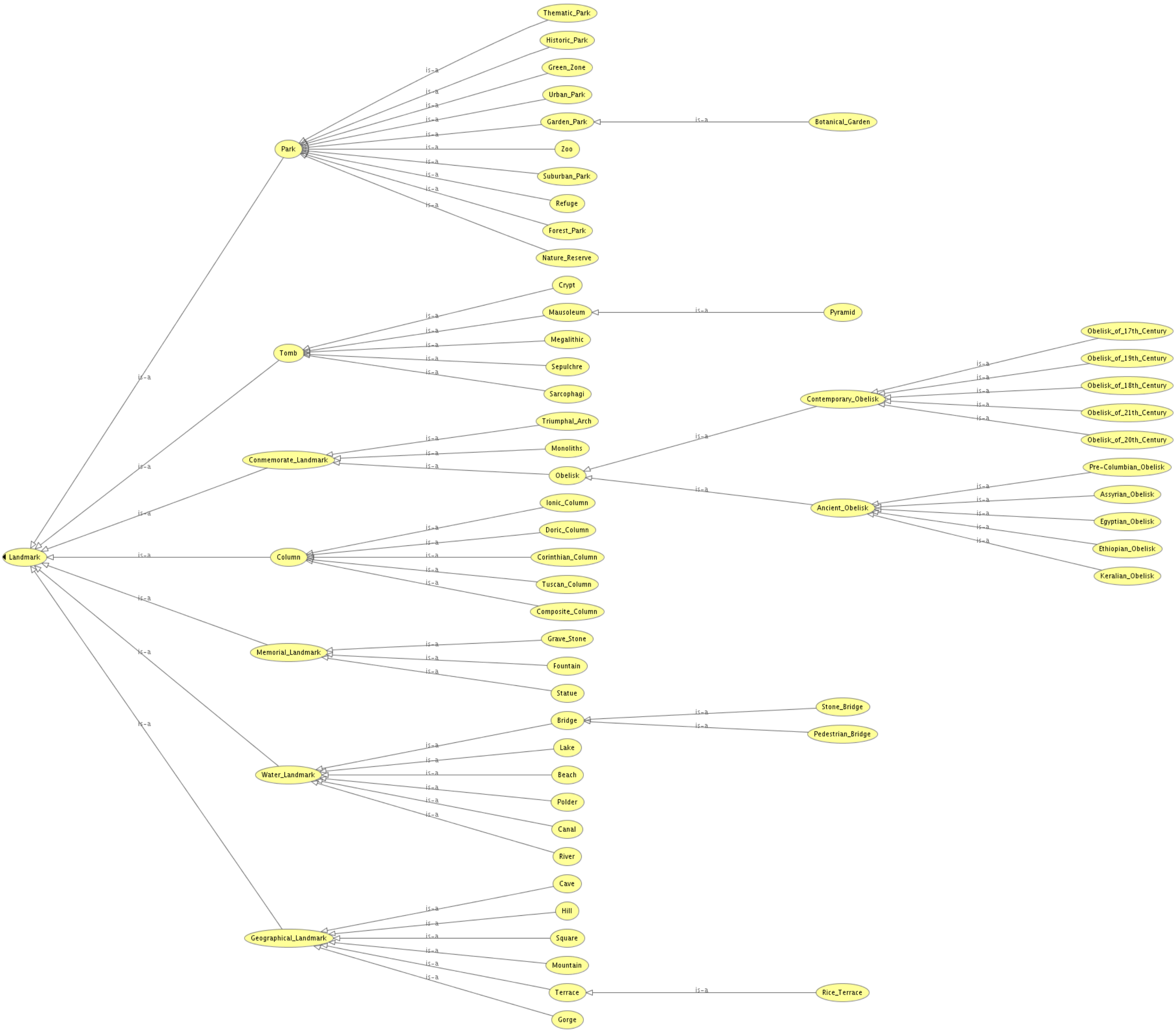
Following it is shown the taxonomy of TourismOWL.owl ontology. Each sub-tree depicts the main classes of the ontology and its hierarchy.


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


Festival
Christian_Festival
Music_Festival
Art_Festival
Song_Festival
Carnival
is-a

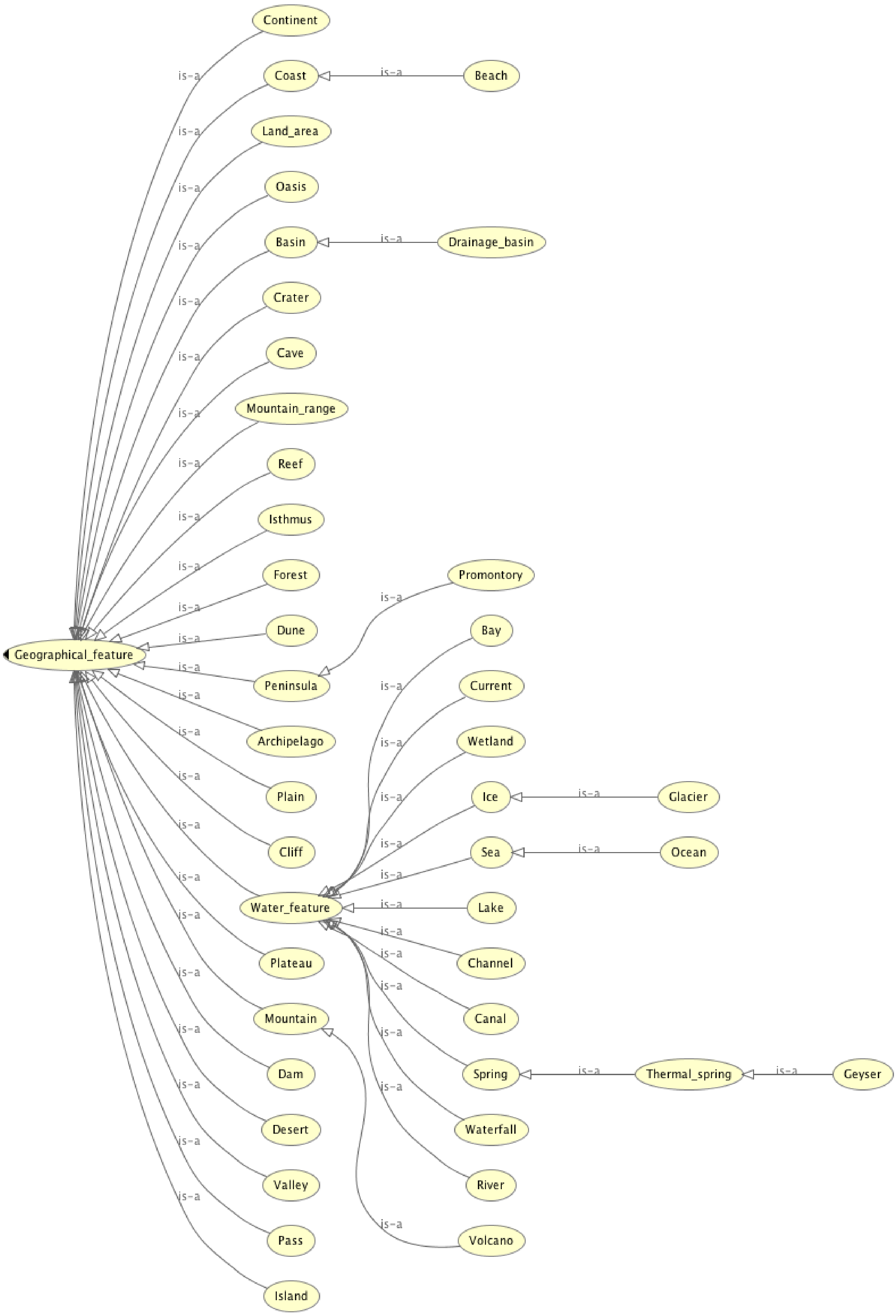

Building
Skyscraper
Industrial_Building
Comercial_Building
Sport_Building
Residential_Building
Cultural_Building
Religious_Building
Militar_Building
Textil_Mill
Cotton_Mill
Fair
Fashion_Fair
Artisan_Fair
Trade_Fair
Shopping
Shopping_Street
Shopping_Area
Shopping_Centre
Mall
Tiangui
Comercial_District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

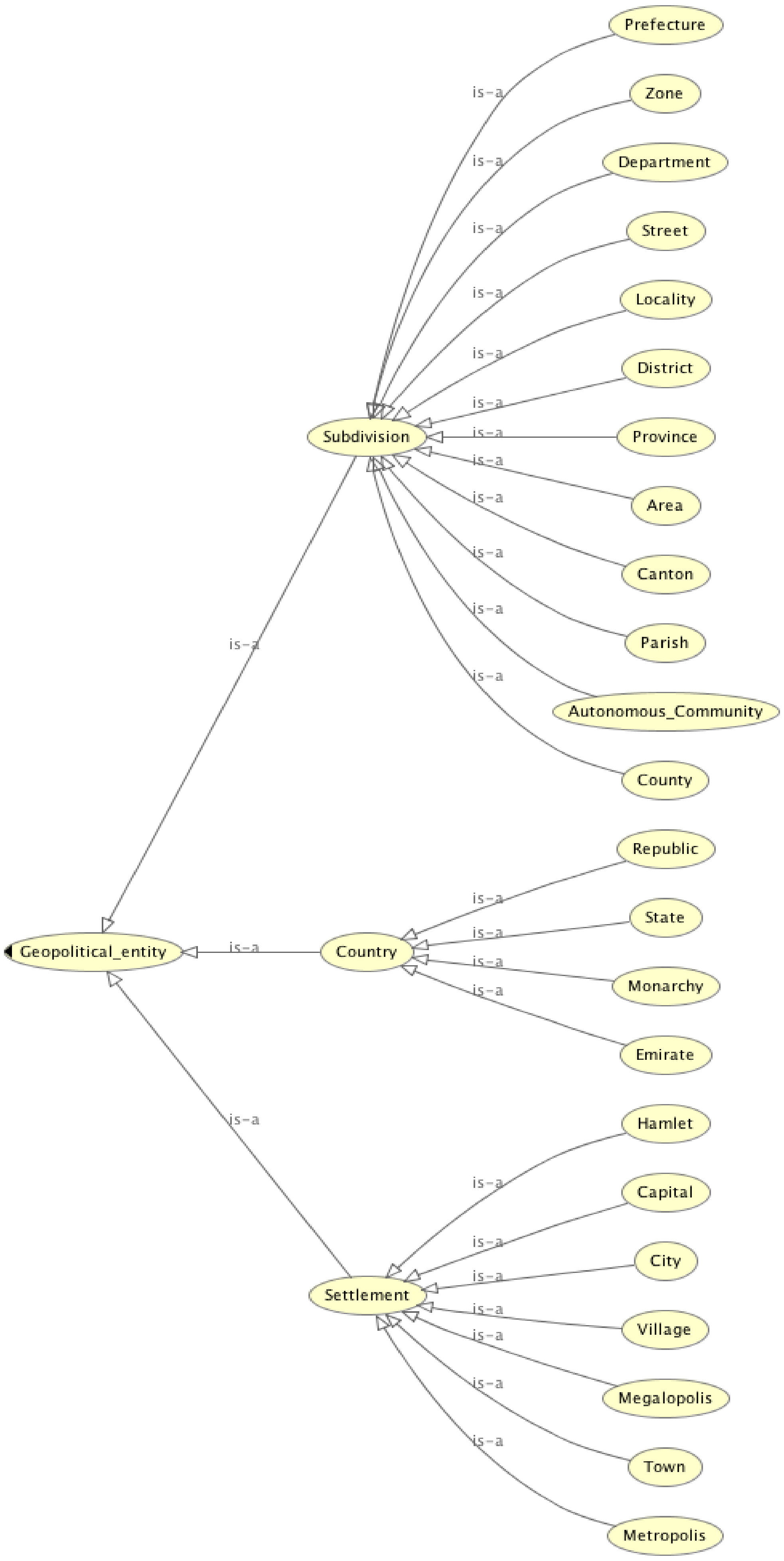

Sport
is-a
Aerial_Sport
Bridge_Jumping
Parachuting
Paraglidgin
Aquatic_Sport
Windsurfing
Rafting
Swimming
Kayaking
Sailing
Diving
Waterskiing
Surfing
Water_Polo
Competition_Sports
Popular_Running
Swimming_Race
Terrestrian_Sport
Street_Sport
Skateboarding
Stickball
Motor_Sport
Motorbike_Sport
Moto_Gp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

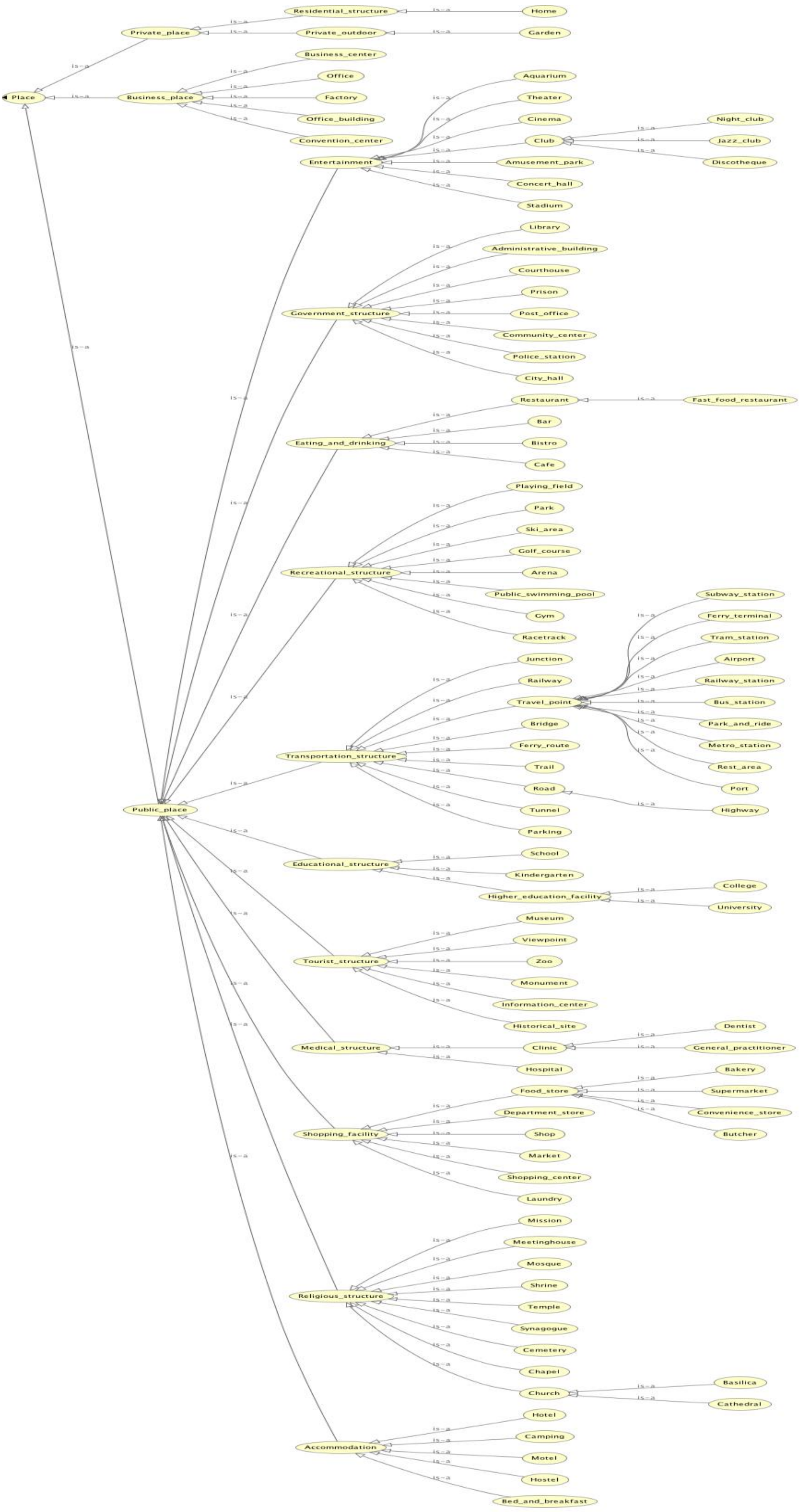

Landmark
is-a
Park
Thematic_Park
Historic_Park
Green_Zone
Urban_Park
Garden_Park
Botanical_Garden
Zoo
Suburban_Park
Refuge
Forest_Park
Nature_Reserve
Tomb
Crypt
Mausoleum
Pyramid
Megalithic
Sepulchre
Sarcophagi
Conmemorate_Landmark
Triumphal_Arch
Monoliths
Obelisk
Contemporary_Obelisk
Obelisk_of_17th_Century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

Annex II – Space.owl

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


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


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


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