TEXT ANNOTATION USING BACKGROUND KNOWLEDGE

Delia Sorina Rusu

Doctoral Dissertation
Jožef Stefan International Postgraduate School
Ljubljana, Slovenia

Supervisor: Prof. Dr. Dunja Mladenić, Jožef Stefan Institute and Jožef Stefan International Postgraduate School, Ljubljana, Slovenia

Evaluation Board:
Asst. Prof. Dr. Tomaž Erjavec, Chair, Jožef Stefan Institute, Ljubljana, Slovenia
Asst. Prof. Dr. Darja Fišer, Member, University of Ljubljana, Faculty of Arts, Ljubljana, Slovenia
Dr. Michael Witbrock, Member, Cycorp, Austin, Texas, United States of America

MEDNARODNA PODIPLOMSKA ŠOLA JOŽEFA STEFANA
JOŽEF STEFAN INTERNATIONAL POSTGRADUATE SCHOOL

Delia Sorina Rusu

TEXT ANNOTATION USING BACKGROUND KNOWLEDGE

Doctoral Dissertation

ANOTACIJA BESEDIL Z UPORABO PREDZNANJA

Doktorska disertacija

Supervisor: Prof. Dr. Dunja Mladenić

Ljubljana, Slovenia, October 2014

To the memory of my grandparents

Acknowledgments

To begin with, I would like to thank my doctoral advisor, Dunja Mladenić, for her guidance and support throughout my studies.

I am grateful to Marko Grobelnik and Blaž Fortuna for their valuable discussions and contributions, and to Carolina Fortuna for introducing me to the Jožef Stefan Institute and to Slovenia.

I would like to thank the members of my doctoral committee, Tomaž Erjavec, Darja Fišer and Michael Witbrock for their insightful comments and remarks.

My gratitude goes to my colleagues at Jožef Stefan Institutes's Artificial Intelligence Laboratory for their contributions to research papers, projects and applications. Special thanks go to Lorand Dali, Alexandra Moraru and Inna Novalija for the time spent together. Thank you Polona Škraba Stanič and Zala Rott Dali for the Slovene abstract translation, and Mateja Zver for all your help .

My dear friends who are spread around the world, I appreciate your effort to keep in touch despite the distances.

Last but not least, I am extremely grateful to my family for their unconditional love and encouragement, and to my partner for his endless patience and understanding.

The research leading to this thesis has received funding from the Slovenian Research Agency and the RENDER (FP7-257790) and XLike (ICT-STREP-288342) European Union projects.

Abstract

The *Semantic Web* aims for the current Web to evolve into a *Web of Data* which can be processed more easily by machines. Achieving this goal involves enriching the existing unstructured data with explicit *semantic information* and *interlinking* the resulting structured data.

As an alternative to explicitly assigning metadata in order to structure plain-text documents, this thesis proposes techniques to *automatically annotate text with background knowledge* defined in ontologies and knowledge bases published as Linked Data. To this end, as a first contribution of this thesis, we define a modular and generic text annotation framework which can use different background knowledge datasets as input. The framework annotates words or collocations (common sequences of words) with corresponding concepts by taking into account the context in which the words or collocations appear. Moreover, the framework does not require additional external semantically-annotated corpora, using only the ontology or knowledge base as both a concept inventory and as a source of information for guiding the annotation process.

The proposed text annotation framework identifies the matching concept for a phrase by relying on the relatedness between concepts. A second contribution of the thesis is the definition of novel concept relatedness measures which take into account different characteristics of the background knowledge dataset: *concept definitions*, i.e. human-readable text describing their meaning, *dataset structure*, which encompasses the various types of relations between concepts and a *hybrid* approach combining the aforementioned characteristics. The concept definition-based measure determines the relatedness between concepts based on a Vector Space Model representation of the definitions, while the structure-based measure relies on a weighting scheme which can quantify the degree of abstractness of concepts.

In order to demonstrate the generality of the proposed approaches, a third contribution of the thesis is the application of the approaches to different cross-domain ontologies and knowledge bases published as Linked Data. The relatedness measures are applied to OpenCyc, WordNet and DBpedia while the text annotation framework links words to concepts from the latter two datasets. OpenCyc is the open source version of the Cyc common-sense knowledge base, WordNet is a well-established lexical database of English and many other languages while DBpedia contains structured encyclopedic information extracted from Wikipedia.

The performance of the concept relatedness and text annotation algorithms is assessed in several evaluation settings. Results show that a hybrid approach which combines concept definitions and the background knowledge dataset structure attains the best results. In the absence of concept definitions, the structure-based relatedness measure is a viable alternative as it closely resembles the human judgment of relatedness. Moreover, the text annotation framework based on the proposed

relatedness measures obtains competitive results for both WordNet and DBpedia evaluations, despite not making use of additional corpora.

Povzetek

Namen semantičnega spleta je nadgradnja trenutnega svetovnega spleta v t. i. splet podatkov, ki bi omogočal lažjo računalniško obdelavo. Doseganje tega cilja zahteva obogatitev obstoječih nestrukturiranih podatkov z eksplicitnimi semantičnimi informacijami ter medsebojno povezovanje tako pridobljenih strukturiranih podatkov.

Namesto strukturiranja navadnih tekstovnih dokumentov z eksplicitnim dodajanjem meta podatkov v doktorskem delu predlagamo alternativne pristope za avtomatsko anotacijo, ki temelji na predznanju, definiranem znotraj ontologij in različnih baz znanja, objavljenih kot Povezan nabor podatkov (angl. Linked Data). V ta namen definiramo modularno in generično ogrodje za anotacijo besedil (angl. text annotation framework), ki lahko kot vhodne podatke uporablja različne baze znanja; to je prvi prispevek tega doktorskega dela. Ogrodje omogoča anotacijo besede ali zaporedja besed z ustreznimi koncepti, tako da upošteva kontekst, znotraj katerega se beseda ali zaporedje besed pojavi. Poleg tega ogrodje ne potrebuje dodatnih zunanjih semantično anotiranih korpusov, ampak uporablja ontologijo ali bazo znanja kot zalogo konceptov in kot vir informacij, ki vodi proces anotiranja.

Predlagano ogrodje za anotacijo besedil identificira ujemajoče se koncepte za dano besedno zvezo na podlagi ujemanja med koncepti. Drugi prispevek doktorskega dela je definiranje izvirnih pristopov za mere ujemanja konceptov, ki upoštevajo različne lastnosti predznanja, podanega v obliki ontologij ali baz znanja: definicije konceptov (npr. ljudem berljiv tekst, ki opisuje pomen koncepta), strukturo, ki obsega različne vrste relacij med koncepti, ter hibridni pristop, ki združuje omenjene lastnosti. Pristop, ki temelji na definiciji koncepta, določa ujemanje med koncepti na podlagi vektorskega prostora reprezentacije definicij, medtem ko pristop, ki temelji na strukturi, uporablja shemo uteževanja, s katero lahko kvantificiramo stopnjo abstraktnosti konceptov.

Uporaba predlaganih pristopov na različnih ontologijah in bazah znanja - ki spadajo v različne domene in so objavljene kot Povezan nabor podatkov -, z namenom prikazati splošnost teh pristopov, je tretji prispevek doktorske disertacije. Različne mere ujemanja konceptov smo uporabili na bazah OpenCyc, WordNet in DBpedia, medtem ko smo avtomatsko anotacijo teksta uporabili za povezovanje besede s koncepti v bazah WordNet in DBpedia. OpenCyc je odprta verzija baze splošnega znanja (angl. common-sense knowledge) Cyc, WordNet je dobro uveljavljena leksikalna podatkovna zbirka angleščine, DBpedia pa vsebuje strukturirane enciklopedične podatke, povzete iz Wikipedije.

Učinkovitost algoritmov za ujemanje konceptov in algoritma za anotacijo besedil smo ocenili pod različnimi evalvacijskimi pogoji. Rezultati kažejo, da je hibriden pristop, ki pri meri ujemanja konceptov vključuje definicije konceptov in strukturo baze znanja, najbolj učinkovit. V primeru, da definicije konceptov niso dostopne, je mera ujemanja na podlagi strukture možna alternativa, saj je zelo podobna človeški

percepciji ujemanja oziroma povezanosti. Ugotovili smo, da predlagano ogrodje za anotacijo besedil, ki temelji na predlaganih merah ujemanja, pri evalvaciji na bazah WordNet in DBpedia dosega primerljive rezultate z že obstoječimi orodji, pri čemer ne potrebuje nobenih dodatnih korpusov.

Contents

List of Figures

List of Tables

List of Algorithms

Abbreviations

HTTP	... Hypertext Transfer Protocol
ICF	... Inverse Concept Frequency
IDF	... Inverse Document Frequency
IRI	... Internationalized Resource Identifier
KB	... Knowledge Base
LCS	... Least Common Subsumer
LDA	... Latent Dirichlet Allocation
LLOD	... Linguistic Linked Open Data
LOD	... Linked Open Data
MDS	... Multidimensional Scaling
NLI	... Natural Language Identifier
NLP	... Natural Language Processing
NLTK	... Natural Language Toolkit
OWL	... Web Ontology Language
RDF	... Resource Description Framework
SemEval	... Semantic Evaluation
SVM	... Support Vector Machines
TF	... Term Frequency
TF-IDF	... Term Frequency, Inverse Document Frequency
TF-ICF	... Term Frequency, Inverse Concept Frequency
URI	... Uniform Resource Identifier
URL	... Uniform Resource Locator

Chapter 1

Introduction

The vast majority of digital information available nowadays, including information published on the Web, is provided as semi-structured or multimedia data (video, audio or images). However, under these conditions, it is particularly hard for machines to process the information content, establish relations between different pieces of information or perform reasoning tasks.

The goal of the *Semantic Web* (Berners-Lee, Hendler, & Lassila, 2001) is to enrich existing data with explicit semantic information, thus making the conversion from a Web of unstructured data to a Semantic Web which machines can process more easily. Several Semantic Web Technologies enable achieving this goal: by explicitly assigning *metadata* to information on the Web, machines can more easily identify and extract this information; *ontologies* providing a *shared understanding* of a domain allow interpreting the extracted information; *logic* is used for processing information, drawing conclusions and explanations for these conclusions (Antoniou & Van Harmelen, 2004). The end result would be a *Web of Data* where structured data is *interlinked*.

Linked Data describes a set of principles for publishing and interlinking structured data on the Web. The basic Linked Data principles outlined in Berners-Lee (2006) are:

- using URIs (Uniform Resource Identifiers) as names for things;
- enabling the lookup of these names by using HTTP (Hypertext Transfer Protocol) URIs;
- using standards like RDF (Resource Description Framework) to provide useful information for a URI;
- including links to other URIs.

Uniform Resource Identifiers (URIs) are a means to identify *resources*, where a resource denotes a thing which can be a document, an abstract concept, etc. (Schreiber & Raimond, 2014). Because URIs are limited to a subset of the ASCII character set, Internationalized Resource Identifiers (IRIs) were proposed as a generalizations of URIs which allow more Unicode characters. The Resource Description Framework (Cyganiak, Wood, & Lanthaler, 2014) is a standard model for representing information on the Web as a set of *{subject, predicate, object} triples* which form an RDF graph. The subject and object are the nodes of the RDF graph while the predicate connects the subject with the object, denoting a relationship. The

direction of the edge is from the subject to the object. The subject is an IRI or a blank node, the predicate is an IRI and the object is an IRI, a literal or a blank node (Cyganiak et al., 2014). Literals are used for strings, numbers or dates while blank nodes represent resources for which IRIs or literals are not provided. An example where all triplet elements are represented by IRIs is the following:

```
<http://dbpedia.org/resource/Copenhagen>
<http://dbpedia.org/ontology/country>
<http://dbpedia.org/resource/Denmark>
```

or informally *{Copenhagen, country, Denmark}*, the subject being *Copenhagen*, the object *Denmark* and the predicate *country*.

Along the years many datasets have been published following Linked Data principles as part of *Linked Open Data (LOD)*, starting with merely 12 datasets at the beginning of 2007 and growing to over 900 datasets seven years later (see Chapter 4). In this thesis we use the term *Linked Datasets* to refer to datasets that are available as Linked Data. Linked Datasets are a largely untapped source of structured information, spanning different domains such as media, geography, publications, life sciences and including several cross-domain datasets. Among the different Linked Datasets part of the LOD, ontologies and knowledge bases are particularly relevant in the context of this thesis. Cross-domain ontologies or knowledge bases such as WordNet (Fellbaum, 2005; Van Assem, Gangemi, & Schreiber, 2006), DBpedia (Lehmann et al., 2014) and OpenCyc (OpenCyc, 2014) are among the largest and most popular sources of structured data published according to Linked Data principles.

Knowledge Bases and Ontologies. Knowledge is formally represented via *conceptualizations*: objects, concepts, entities from an area of interest and the relationships between them (Genesereth & Nilsson, 1987). Knowledge bases store this representation, enabling computer systems to access it in an efficient manner. Some knowledge bases such as Cyc (Lenat, 1995) are created and maintained by a group of knowledge engineers while other knowledge bases such as DBpedia or WikiData (Vrandečić & Krötzsch, 2014) are collaborative, their content being created and maintained by numerous contributors. Ontologies explicitly specify conceptualizations, usually from a specific domain, as a set of concepts and relationships between concepts, where the possible interpretations of concepts are constrained by formal axioms (Gruber, 1995). *Concepts* are formally described via *classes*, where a class may have several specific *instances*. In some cases classes and instances are associated human-readable text describing their meaning.

This thesis addresses the problem of *automatically annotating text with background knowledge* defined in ontologies and knowledge bases published as Linked Data, as an alternative to explicitly assigning metadata in order to structure information. This approach has several advantages. First, by considering Linked Data as a source of background knowledge we can propose a solution which is not tailored to a specific ontology or knowledge base. This is because the datasets published as Linked data share some basic characteristics outlined in (Berners-Lee, 2006): a) using URI or IRI references to identify concepts and relations, b) uniformly querying resources based on a common, graph-based data model (RDF) which enables an easier integration of resources, c) using RDF links to connect resources. Due to these basic characteristics, the algorithms presented in this thesis can be applied

to other datasets, not exemplified in this thesis, but which are also published as Linked Data. Second, text information would be structured and interlinked, thus easier to process, understand and reason about. By annotating a word in text with a concept defined in one Linked Dataset, we can also obtain interlinked concepts from other Linked Datasets. This additional structured information could be made available either directly to end-users or to other applications that further process and integrate it. Third, by establishing the link between concepts defined in ontologies and unstructured text we can obtain machine readable representations of text at different levels of granularity; linking to instances offers a more fine-grained view while linking to upper-level ontology classes enables a more abstract representation. Fourth, structured representations of text which take semantics into account can replace the commonly-used bag-of-words text representation in a series of applications such as information extraction, question answering, summarization or machine translation.

We split the text annotation problem into two main subproblems, and start with *determining the degree of relatedness between concepts* defined in ontologies and knowledge bases. Next, we propose a *generic framework for text annotation using background knowledge* which relies on the relatedness between concepts. As a source of background knowledge we focus on three popular cross-domain ontologies and knowledge bases which are part of Linked Open Data: WordNet, OpenCyc and DBpedia. WordNet (Fellbaum, 2005; Van Assem et al., 2006) is a well-known lexical database of English, OpenCyc (OpenCyc, 2014) is the open source version of the Cyc common-sense knowledge base and DBpedia contains structured information extracted from the Wikipedia encyclopedia (Wikipedia, 2014).

In what follows we briefly describe the two subproblems, motivating and connecting them to some of the most relevant existing research. Chapter 2 provides further details regarding related research.

Concept Relatedness. An important task with a long research history and multiple application domains is that of determining the degree of similarity and relatedness between concepts defined in knowledge bases and ontologies. Semantic similarity and relatedness between concepts reflect how closely associated concepts are. Similarity is determined based on the super-subordinate relation - *hypernymy*, *hyponymy* or *IS-A* relation. Relatedness, on the other hand, is not restricted to the super-subordinate relation, and includes other relations such as *part-whole* relations - *meronymy* or *PART-OF*. For example, the concepts *desktop computer* and *tablet computer* are similar as they both refer to a type of computer while the concepts *desktop computer* and *keyboard* are related as the keyboard can be part of the desktop computer.

There are numerous applications which take advantage of the similarity or relatedness between concepts. In a word sense disambiguation setting, knowing how similar concepts are enables identifying the corresponding set of concepts which match a phrase in a given context (Navigli, 2009). Euzenat and Shvaiko (2007) show that two ontologies can be aligned based on the elements they have in common. Concept similarity can also improve the search engine results in information retrieval applications (Hliaoutakis, Varelas, Voutsakis, Petrakis, & Milios, 2006), as well as learning based on knowledge sources using different machine learning approaches, e.g. clustering or classification (Milne & Witten, 2013). Another application domain is biomedical and geo-informatics, where concept similarity can be used to compare genes and proteins (The Gene Ontology Consortium, 2000) and geographic features,

respectively.

For assessing the similarity or relatedness between concepts, several external knowledge sources have been utilized: thesauri, which define relationships between words, machine readable dictionaries such as the Collins English Dictionary (Collins English Dictionary, 2014), domain-specific ontologies such as the Gene Ontology (The Gene Ontology Consortium, 2000) or more generic ontologies such as Cyc or DBpedia. The WordNet lexical database and its extensions can be arguably viewed as an ontology including a taxonomy of concepts and a set of semantic relations defined between them. WordNet is also used in evaluating different similarity and relatedness measures under a common setting, and it is one of the most utilized knowledge sources.

Cognitive psychology proposes different theoretical models of similarity and relatedness:

- geometric models for representing concepts and the relationships between them, notably Quillian's model of semantic memory (Quillian, 1968);
- the feature matching model where concepts are described by a set of features or attributes (Tversky, 1977).

Based on these models, researchers have described a number of approaches to measuring similarity and relatedness. A very popular direction was exploiting the WordNet network of semantic connections (Rada, Mili, Bicknell, & Blettner, 1989; Sussna, 1993; Agirre & Rigau, 1996; Leacock & Chodorow, 1998). Other approaches were based on the distance – i.e. the number of semantic connections - between concepts (Rada et al., 1989; Wu & Palmer, 1994; Leacock & Chodorow, 1998). Resnik (1995) proposed a measure based on information content - i.e. on the probability of occurrence of a concept. Pirro and Euzenat (2010) applied a feature-based model in an information theoretic framework. Semantic similarity was also defined in Description Logics (Janowicz & Wilkes, 2009).

We identify a number of challenges in determining the similarity and relatedness between concepts defined in ontologies and knowledge bases when utilizing state-of-the-art algorithms. These challenges are rooted in the fact that ontologies and knowledge bases can differ in structure, way of specifying conceptualizations, and information provided for each concept. Firstly, methods that provide good results for a given ontology or knowledge base turn out to perform poorly on another one. For example, WordNet-based measures that take into account concept definitions do not produce equally good results when applied to other ontologies such as OpenCyc (Rusu, Fortuna, & Mladenić, 2011). Secondly, information content-based measures rely on the probability of occurrence of a concept. These probabilities can be inferred from frequencies of words in external corpora; however, in this case the polysemy of words or phrases is not taken into account (see Section 2.1.3). Moreover, word frequencies and concept frequencies are not equivalent. An alternative is to infer concept probabilities based on semantically-annotated corpora such as SemCor (Landes, Leacock, & Tengi, 1998); the drawback is that such corpora are expensive to obtain. Different application domains, however, require different corpora. Thirdly, methods that are based on the distance between concepts treat all semantic connections between concepts uniformly. Additionally, these methods interpret the distance between more specific and more abstract concepts in the same manner. This is not appropriate for most ontologies, as a short distance between

two concepts, determined based on the number of relations separating the concepts, does not necessarily imply that the concepts are semantically close (Pirro & Euzenat, 2010). For example the concept pairs *entity* - *thing* and *bicycle* - *wheel* are not equally close semantically, even if the distance in both cases is equal to one.

Text Annotation. Annotating text with concepts defined in ontologies or knowledge bases can also be seen as a *word sense disambiguation* task, one of the oldest computational linguistics problems dating back to the 1940s. Word sense disambiguation involves the identification of the meaning of words in a given context based on an inventory of senses (Navigli, 2009). Similarly, we annotate text with ontological concepts by selecting the most appropriate concept from a number of candidate concepts.

Three main approaches have emerged for text annotation: *supervised*, *unsupervised* and *knowledge-based*. Supervised techniques which employ machine learning methods for training a classifier on concept-labeled data have obtained the most promising results. However, these algorithms require annotated data and need retraining for other domains or languages. Moreover, they are expensive to train or operate on a broader scale due to the scarcity of labeled data. These drawbacks brought about unsupervised techniques, relying on clustering of word contexts, and knowledge-based approaches which exploit various concept inventories like dictionaries, ontologies or thesauri to determine the appropriate concept for a given word in context. As opposed to supervised methods, unsupervised techniques require no training, have wider coverage and are easier to adapt to other domains or languages while providing lower quality results. Knowledge-based approaches share the advantages of unsupervised techniques and in addition benefit from the linguistic and semantic information encoded in the knowledge base. Yet the coverage and quality of this type of approach depends on the quality of the underlying knowledge base. Hybrid systems may use weakly supervised techniques which leverage seed data or unsupervised methods based on cross-lingual evidence (Navigli, 2009).

Moving closer to real-world applications involving the annotation of domain-specific and multilingual datasets, the challenges are threefold. Firstly, most of the annotation algorithms have been developed having in mind a particular knowledge base, the most popular ones being WordNet and Wikipedia. However, few of the proposed algorithms are generic enough to be applied to other ontologies or knowledge bases than the ones they were initially designed for. Secondly, some text annotation systems are based on domain-specific annotated corpora, which is expensive to obtain (Agirre et al., 2010). Thirdly, multilingual text annotation implies either language-agnostic algorithms or the availability of language-dependent tools such as named entity recognizers or parsers for the target language.

1.1 Terminology

The topic of this thesis is *automatic text annotation using background knowledge*. In the context of this thesis, *text annotation* involves identifying suitable concepts for words or collocations by taking into account the context in which the words or collocations appear. *Collocations* are sequences of words which co-occur with a frequency that is significantly higher than what would be expected under the assumption of independent occurrences. An example collocation is *strong tea*. In the sentence "The two boys are good friends." we would annotate the word "boys" with a concept denoting a *young male person* or the word "friends" with a concept

representing the meaning of *a person whom one knows well*. As an intermediary step the words "boys" and "friends" are lemmatized, and the corresponding lemmas or base forms "boy" and "friend" are matched to WordNet concepts.

There have been numerous research efforts directed at building structured knowledge sources such as machine readable dictionaries, knowledge bases and ontologies. We refer to these structured knowledge sources as *background knowledge*, which we use as a *concept inventory*. Coming back to our example sentence, if we used WordNet 3.0 as the concept inventory, we could choose among several concepts in order to annotate the words "boys" and "friends", respectively (see Table 1.1). The concepts which represent possible annotations for a given word or collocation are called *candidate concepts*. In this example there are three candidate concepts for the word "boys" and five for the word "friends". The mapping between words and concepts can be achieved via the concept *Natural Language Identifiers (NLI)*. In Table 1.1 the NLIs have been marked in bold, and the matching NLI has been underlined.

The WordNet concepts which can be associated with the words "boy" and "friend", respectively, are represented via ontology instances of the *NounSynset* class. In OpenCyc, on the other hand, the word "friend" would be mapped to the object property *friends*, while the word "boy" would be mapped to the OpenCyc class *Boy*.

Table 1.1: Concepts corresponding to the words "boy" and "friend" in WordNet 3.0. The natural language identifiers have been marked in bold, and the matching NLI has been underlined.

1. **male child**, **boy** - a youthful male person	1. **friend** - a person you know well and regard with affection and trust
2. **boy** - a friendly informal reference to a grown man	2. **ally**, **friend** - an associate who provides cooperation or assistance
3. **son**, **boy** - a male human offspring	3. **acquaintance**, **friend** - a person with whom you are acquainted
	4. **supporter**, **protagonist**, **champion**, **admirer**, **booster**, **friend** - a person who backs a politician or a team etc.
	5. **Friend**, **Quaker** - a member of the Religious Society of Friends founded by George Fox

One approach to identifying which of the candidate concepts best matches the word in context is to determine the *relatedness* between concept pairs. For our example we would obtain 15 relatedness pairs for the words ("boys", "friends"): $(boy_1, friend_1)$, $(boy_1, friend_2) \ldots (boy_3, friend_5)$, where boy_i and $friend_j$ represent the senses of these words in WordNet. The pairs can be ranked based on their corresponding pairwise relatedness value, providing an indication of which pair(s) of concepts is most suitable for annotating words in the example sentence.

Concept relatedness can be determined based on different characteristics of the ontology or knowledge base. Algorithms can use the *concept definition*, i.e. human-readable text describing the meaning of the concept or take into account different *relations between concepts*. In the aforementioned example, the definition associated with the concept boy_3 is *a male human offspring*; this concept is connected to several other concepts via different types of relations, for example the concept *Junior*

defined as *a son who has the same first name as his father* is one of its *hyponyms*.

1.2 Aims and Hypothesis

The general aim of this dissertation is to *propose, apply and evaluate a generic text annotation framework based on background knowledge datasets, using Computational Linguistics and Semantic Technologies*. This aim is further broken down into the following items:

- Define algorithms for determining the relatedness between concepts represented in background knowledge datasets part of Linked Open Data. These algorithms take into account different properties of the background knowledge datasets: concept definitions, dataset structure and a hybrid algorithm which combines the aforementioned two approaches. These algorithms are presented in Chapter 3.

- Define a generic text annotation framework using background knowledge, which integrates different concept relatedness algorithms. The annotation framework is described in Chapter 5.

- Apply the concept relatedness algorithms and the text annotation framework to several background knowledge datasets with different properties. The cross-domain datasets used for exemplification, namely WordNet, OpenCyc and DBpedia are presented in Chapter 4.

- Evaluate the relatedness algorithms and the text annotation framework as a whole, using different background knowledge datasets. The evaluation settings are described in Chapter 6.

In this thesis we address two hypotheses that we test experimentally:

1. *Common background knowledge dataset characteristics enable us to define generic concept relatedness measures and a text annotation framework based on these measures which are applicable to different datasets.*

   We evaluate the generality of our approach by applying the relatedness measure in the case of three cross-domain Linked Datasets: WordNet, OpenCyc and DBpedia, while the text annotation framework is applied to WordNet and DBpedia, respectively (see Chapter 6 for the evaluation results).

2. *Algorithms that take into account different types of information provided by the background knowledge datasets outperform the algorithms that are based on a single type of information.*

   In order to test this hypothesis we propose three types of relatedness measures which we integrate in the text annotation framework. These measures rely on concept definitions, dataset structure and a hybrid algorithm which combines the aforementioned two approaches. The performance of these approaches is tested on datasets having different characteristics (see Chapter 6).

1.3 Scientific Contributions

The main contributions of this thesis are *the definition, application and evaluation of a generic text annotation framework using background knowledge which integrates different concept relatedness algorithms.* The scientific relevance of the thesis lies on the applicability of the proposed algorithms to other datasets, not exemplified in the thesis, provided these other datasets share some basic properties with the exemplified datasets. We claim the following contributions to the fields of Computational Linguistics and Semantic Web:

- Proposing novel approaches to determine the relatedness between concepts defined in background knowledge datasets such as ontologies and knowledge bases. The relatedness measures leverage concept definitions, the background knowledge dataset structure as well as a combination of concept definitions and dataset structure.

- Defining a modular and generic automatic text annotation framework which relies on the relatedness between concepts. The framework annotates words and collocations in a text fragment with concepts represented in a background knowledge dataset and does not require additional external semantically-annotated corpora.

- Applying and evaluating the relatedness measures and the text annotation framework in the case of several background knowledge datasets with different characteristics: WordNet, OpenCyc and DBpedia, in order to show the generality of the proposed approaches.

First, this thesis proposes novel approaches to determine the relatedness between concepts defined in background knowledge datasets, which rely on different dataset characteristics. The concept definition-based measure uses a Vector Space Model to represent concept definitions; the relatedness between concepts is obtained via a kernel function which leverages the contribution of different concept definitions. The structure-based measure relies on the geometric representation of concepts and their mutual relationships. We distinguish concepts based on their degree of abstractness (Resnik, 1995) and describe a weighting scheme which can quantify this degree of abstractness. The relatedness algorithm is based on the notion of shortest path, as defined in graph theory. A hybrid measure combines the concept definition-based measure and the structure-based measure.

Second, we define a modular yet generic text annotation framework which can be applied to assign concepts to words in a text fragment using different background knowledge datasets as input. The text annotation framework relies on the relatedness between concepts defined in the input dataset, and does not require external corpora which are semantically annotated. In such corpora words in context are tagged with concepts from a concept inventory. Even though we focus on annotating English text, our approach is language independent and can be used to annotate text in other languages, provided there exists an ontology or knowledge base for that language.

Third, we use different background knowledge datasets part of Linked Open Data (WordNet, OpenCyc and DBpedia) in order to show the applicability and generality of the proposed relatedness measures and the text annotation framework. In general

we obtain best results for both concept relatedness and text annotation tasks when combining the information provided by concept definitions with the background knowledge dataset structure. For ontologies or knowledge bases such as OpenCyc where less than half of the concepts have assigned a definition we show that the proposed structure-based method obtains competitive results.

The implementations of the algorithms proposed in this thesis have been open-sourced and are available at `https://github.com/deliarusu/text-annotation.git`. This enables researchers to apply them in the case of other background knowledge datasets or to integrate them in different text annotation or analysis frameworks. Appendix A contains a description of the implementation.

1.4 Thesis Structure

In this chapter we described the research area which constitutes the focus of this thesis, and presented the terminology used in the thesis. Next we pointed out the main aims and hypotheses and highlighted the scientific contributions claimed in the thesis. The remainder of the thesis is structured as follows.

Chapter 2 provides an overview of the related work, focusing on measures of semantic similarity and relatedness and text annotation approaches, respectively. We present different similarity and relatedness measures and text annotation algorithms and their application to various knowledge bases used as background knowledge.

Chapter 3 proposes three measures of relatedness between concepts, taking into account concept definitions, the knowledge base structure and a hybrid approach which is a combination of the two types of measures.

In Chapter 4 we describe the linked datasets used as background knowledge: WordNet, OpenCyc and DBpedia. For each case we give an overview of the knowledge base, we explain how to identify candidate concepts for words or collocations to be annotated and we provide an illustrative example.

Chapter 5 presents one of the main contributions of this thesis, namely the *Automatic Text Annotation Framework* which integrates the concept relatedness measures and relies on a knowledge base as a source of background knowledge.

The proposed algorithms are evaluated in Chapter 6. We start by evaluating the relatedness measures on standard datasets (for WordNet, OpenCyc and DBpedia) and synthetic data (in the case of OpenCyc), and continue with text annotation experiments using WordNet and DBpedia as knowledge bases. A discussion of the results follows in Chapter 7.

The final chapter of this thesis (Chapter 8) includes concluding remarks and proposes future work directions.

Chapter 2

Related Work

This chapter provides an overview of related work regarding measures of semantic similarity and relatedness and text annotation approaches which use different background knowledge datasets, the most popular being WordNet and Wikipedia.

2.1 Measures of Similarity and Relatedness

Concept similarity and relatedness have been extensively analyzed within computational linguistics research. Semantic similarity and relatedness reflect the strength of the relation between concepts. If the relation is restricted to a super-subordinate one, we talk about *concept similarity*, otherwise, for the more general case, we use the term *concept relatedness*. Most of the proposed methods for determining concept similarity and relatedness have been developed and tested for the WordNet English lexical database. Validating and comparing different approaches is part of ongoing research. So far the evaluation involves comparing the proposed method against a manually-created golden standard where word pairs are given a score reflecting how related they are. Yet available golden standards are limited to merely tens (Rubenstein & Goodenough, 1965; Millers & Charles, 1991) or hundreds (Finkelstein et al., 2010) of word pairs. More recently Paulheim (2013) has released a machine generated silver standard for DBpedia resources, consisting of almost 7,000 pairs of resources.

In what follows, we present some of the most cited approaches, which rely on different characteristics of the ontology or knowledge base. We start by describing concept definition-based algorithms. They incorporate concept-related information into the similarity measure, e.g., concept "dictionary-like" definitions or various labels attached to the concepts. As not all ontologies have definitions associated to the concepts, the second type of algorithms – structure-based algorithms – take into account the ontological structure. In some cases the similarity measure incorporates both the concept definitions, as well as the structure of the ontology. Another category of approaches is the information theoretic one. Central to this group of approaches is the notion of information content. In this case concepts are assigned probabilities based on word frequencies in text corpora such as the Brown Corpus of American English (Francis, Kučera, & Mackie, 1982).

2.1.1 Definition-based Measures

In this section we present existing concept-based algorithms, derived from the well-established Lesk algorithm.

Lesk algorithm and its extensions. Definition overlap or the *Lesk algorithm* (Lesk, 1986) is based on computing the overlap between two or more concept definitions, where the concepts belong to a concept inventory such as a knowledge base or ontology. Each word in a given text fragment is assigned several candidate concepts from the concept inventory. The candidate concepts are selected using various techniques, the most straightforward being string matching between the word in text and the concept natural language identifier. The initial Lesk algorithm computes the overlap between the concept definitions as follows. Given two concepts c_1 and c_2, the similarity between the two concepts is determined by counting the number of common words in the definitions of the two concepts:

$$Similarity_{Lesk}(c_1, c_2) = |definition(c_1) \cap definition(c_2)| \tag{2.1}$$

An extended version of the algorithm, called *Extended Definition Overlap* (Banerjee & Pedersen, 2003) takes into account, in addition to the definitions of the two concepts, definitions of related concepts. Examples of related concepts are hypernyms, meronyms, etc. Thus, this algorithm considers both the concept definitions, as well as the structure of the ontology.

Definition Vectors. Patwardhan and Pedersen (2006) create second order co-occurrence vectors from concept definitions, called *definition vectors*. The authors define a *Word Space* which includes all words in WordNet concept definitions, except stop words and infrequent words (occurring below a certain threshold). For every such word w a first order *context vector* $\overrightarrow{w}$ is created by incrementing the dimensions of $\overrightarrow{w}$ for co-occurrences of w. The definition vector of a concept is therefore obtained by summing the first order context vectors from the concept definition. The similarity between two concepts is defined as the cosine similarity between the corresponding definition vectors.

The measures proposed in (Banerjee & Pedersen, 2003; Patwardhan & Pedersen, 2006) make use of other types of relations in addition to the subsumption one, and are therefore considered relatedness measures.

2.1.2 Structure-based Measures

Structure-based measures view the ontology as a graph where nodes represent the concepts and the graph edges stand for the relationships between concepts. On this graph measures for distance (minimum for identical concepts) or similarity (maximum for identical concepts) can be defined. Graph theory literature discusses numerous node and edge weighting schemes, as well as algorithms based on these schemes. In his work on similarity in knowledge graphs, Hoede (1986) compared the in-degrees and out-degrees of two nodes in order to determine how similar these nodes are. Moore, Steinke, and Tresp (2011) have previously used node degrees to define edge weights and identify paths in DBpedia and OpenCyc. Their purpose was to determine relevant neighbors for a given query node, and further to discover

interesting links between two given nodes. Given the edge weights, we can apply a standard graph algorithm for identifying the shortest path between two nodes. One such algorithm is the Dijkstra algorithm (Dijkstra, 1959).

In what follows, we present the most common measures.

Shortest Path. Rada et al. (1989) introduce a simple measure for the distance between two concepts; it is obtained by counting the number of edges in the shortest path between the concepts:

$$Similarity_{ShortestPath}(c_1, c_2) = \text{minimum number of edges separating } c_1 \text{ and } c_2 \tag{2.2}$$

The authors see this conceptual distance as a decreasing function of similarity, i.e. the smaller the conceptual distance, the more similar the concepts. They initially computed the shortest paths on the WordNet and MeSH (MeSH, 2014) taxonomies. MeSH (Medical Subject Headings) is a hierarchy of medical and biological terms.

Rada et al. show that by representing concepts as points in a multidimensional space, the conceptual distance can be measured by the geometric distance between the points. The distance metric is defined based on Quillian's spreading-activation theory (Quillian, 1968). According to this theory, memory search is viewed as activation spreading in a semantic network. The aim is to recreate the human brain's semantic structure and parallel processing capability via a standard (serial processing) computer (Collins & Loftus, 1975). Quillian's model of semantic memory consists of nodes and links between them. The memory nodes represent concepts, whereas the links represent the relationships between concepts. The semantic memory is organized such that nodes that represent closely related concepts have many links between them. Quillian assigns *criteriality tags* to links in order to show the strength of the link. The spreading activation theory stipulates that two concepts can be compared by tracing the paths between their corresponding nodes. Depending on the criteriality tags of the links in these paths, the concepts are considered to be more or less similar.

Rada et al.'s work emphasizes the fact that the distance metric is mainly designed to work with hierarchical knowledge bases. Moreover, in the model of semantic memory that the distance metric is based on, the super-subordinate relation *IS-A* is assigned a high criteriality tag, signifying its importance. The main drawback of the distance metric is that it assumes more specific and more abstract concepts to have the same interpretation, which is not valid in most knowledge bases (Resnik, 1995). However, overcoming this drawback is not straight-forward, as different ontologies or knowledge bases have very different approaches to defining the concept hierarchy. Take for example WordNet and OpenCyc. WordNet is a lexical database where the concepts cover the common English lexicon. OpenCyc, on the other hand, is a common-sense knowledge base primarily developed for modeling and reasoning about the world. As such, it contains various abstract concepts, e.g. *Collection* is an OpenCyc concept representing *the collection of all collections of things. Each Collection is a kind or type of thing whose instances share a certain property, attribute, or feature.*

Leacock and Chodorow. Another structure-based similarity measure using the distance between two concepts is proposed in (Leacock & Chodorow, 1998).

In this case, the shortest path between two concepts is scaled by the depth of the taxonomy, D.

$$Similarity_{LeacockChodorow}(c_1, c_2) = \max_i \left[-\log \frac{N_{p_i}}{2 \cdot D} \right], \tag{2.3}$$

where N_p is the number of nodes in path p from c_1 to c_2.

Wu and Palmer. This measure (Wu & Palmer, 1994) relies on determining the depth of concepts in a taxonomy, i.e. counting the number of concepts in the path between a concept and the root concept, taking into account the *Least Common Subsumer (LCS)* of the two concepts. In a taxonomy such as WordNet, the Least Common Subsumer is the closest common ancestor of the two concepts c_1 and c_2.

$$Similarity_{WuPalmer}(c_1, c_2) = \frac{2 \cdot N_3}{N_1 + N_2 + 2 \cdot N_3}, \tag{2.4}$$

where N_1 is the number of nodes in the path from c_1 to the $LCS(c_1, c_2)$, N_2 is the number of nodes in the path from c_2 to the $LCS(c_1, c_2)$ and N_3 is the number of nodes in the path from the $LCS(c_1, c_2)$ to the root of the taxonomy.

Several relatedness measures have been proposed and validated using the WordNet ontology.

Lexical Chains. Hirst and St-Onge (1998) describes a relatedness measure defined between two concepts which is centered on the idea of semantically correct paths described by a set of rules. Each relation type is associated with a direction: **U**pward, **D**ownward and **H**orizontal. The upward link corresponds to generalization, the downward to specialization and the horizontal link corresponds to relations such as antonymy, similarity, see also. Given the set of rules, the authors identify eight patterns of semantically-correct paths: *{U, UD, UH, UHD, D, DH, HD, H}*. The same idea of semantically correct paths is further extended in (Mazuel & Sabouret, 2008). The types of relations are limited to hierarchical ones and object properties. In this work, the assumption that "two different hierarchical edges do not carry the same information content" is extended to non-hierarchical links.

Yang and Powers. Yang and Powers (2006) propose a relatedness measure defined between two concepts which relies on an edge-based counting model where edges are weighted depending on their type. The authors analyze two main relationship types: *IS-A* and *PART-OF*.

2.1.3 Information Content-based Measures

Resnik. A semantic similarity measure for taxonomies, based on the notion of information content, is proposed in (Resnik, 1995). The concepts in the taxonomy are associated with a probability of occurrence estimated using noun frequencies from the Brown Corpus of American English. This corpus provides word frequencies in a collection of texts belonging to different genres ranging from news articles to science fiction. The frequency of a concept $freq(c)$ is determined based on noun frequencies:

$$freq(c) = \sum_{n \in words(c)} count(n), \tag{2.5}$$

where n is a noun and $words(c)$ represents the set of words subsumed by c. For example, an occurrence of the word "bicycle" would increase the frequency of the concepts *bicycle*, *mountain bike*, *velocipede*, etc.

Concept probabilities are relative frequencies:

$$\hat{p}(c) = \frac{freq(c)}{N}, \tag{2.6}$$

where N represents the total number of concepts. This is a rough estimate for the probability of a concept, and does not take into account word polysemy.

The more abstract a concept is, the lower its information content. The information content IC of a concept c is defined as:

$$IC(c) = -\log(p(c)), \tag{2.7}$$

The semantic similarity proposed by Resnik is defined as follows, where $S(c_1, c_2)$ is the set of concepts subsuming both c_1 and c_2.

$$Similarity_{Resnik}(c_1, c_2) = \max_{c \in S(c_1, c_2)} [IC(c)] \tag{2.8}$$

Jiang and Conrath. Jiang and Conrath (1997) use the notion of information content as a decision factor in a model derived from the edge-based notion proposed in (Rada et al., 1989). They define the following distance function between two concepts:

$$Distance_{JiangConrath}(c_1, c_2) = IC(c_1) + IC(c_2) - 2 \cdot IC(LCS(c_1, c_2)), \tag{2.9}$$

where LCS denotes the Least Common Subsumer.

Lin. A different version of the Jiang and Conrath distance is described in (Lin, 1998):

$$Similarity_{Lin}(c_1, c_2) = \frac{2 \cdot IC(F(c_1) \cap F(c_2))}{IC(F(c_1)) + IC(F(c_2))}, \tag{2.10}$$

where $F(c)$ represents the set of features of concept c.

Intrinsic and Extended Information Content. Instead of utilizing external corpora to determine concept probabilities, Seco, Veale, and Hayes (2004) introduce the *Intrinsic Information Content*, where the probability of a concept is estimated based on the number of hyponyms of that concept:

$$IC_{WordNet}(c) = 1 - \frac{\log(hypo(c) + 1)}{\log(max_{WordNet})}, \tag{2.11}$$

where $hypo(c)$ represents the number of hyponyms for the concept c while $max_{WordNet}$ are the number of WordNet taxonomy concepts.

This formulation is extended to take advantage of all ontological relations existing between concepts, resulting in the *Extended Information Content* (Pirro & Euzenat, 2010). The Extended Information Content $eIC(c)$ is defined as a weighted sum of the Intrinsic Information Content $iIC(c)$ and a coefficient $EIC(c)$. The EIC coefficient takes into account all m relations between the concept c and other concepts in the ontology.

$$EIC(c) = \sum_{j=1}^{m} \frac{\sum_{k=1}^{n} iIC(c_k \in C_{R_j})}{|C_{R_j}|} \tag{2.12}$$

$$eIC(c) = \zeta iIC(c) + \eta EIC(c) \tag{2.13}$$

Together, Intrinsic and Extended Information Content are used in a framework inspired from Tversky's feature-based model (Pirro, 2009; Pirro & Euzenat, 2010). Intrinsic and Extended information content-based measures have been applied in the cases of WordNet and MeSH (Seco et al., 2004; Pirro & Euzenat, 2010), as well as to determine semantic similarity in biomedical ontologies (Pesquita, Faria, Falcao, Lord, & Couto, 2009).

2.1.4 Wikipedia-based Relatedness Measures

Wikipedia has also been a popular knowledge base used in the semantic relatedness task.

WikiRelate!. Strube and Ponzetto (2006) adapt some of the most popular measures developed for the WordNet lexical database in order to determine the semantic relatedness of concepts represented by Wikipedia pages. They apply text overlap measures to Wikipedia article pages and path and information content-based measures to the Wikipedia category graph.

Explicit Semantic Analysis. Gabrilovich and Markovitch (2007) determine the relatedness between two text fragments by comparing their *semantic interpretation* vectors using a cosine metric. As a first step, the text fragment is represented as a bag of words weighted using the $TF - IDF$ scheme (Manning, Raghavan, & Schütze, 2008). Next, an inverted index maps words to Wikipedia concepts given that each Wikipedia concept is represented as a vector of words from the corresponding Wikipedia article, weighted using the $TF - IDF$ scheme. Finally the *semantic interpretation* vector is composed of weighted Wikipedia concepts corresponding to words in the input text T. The weight of each concept c_j is:

$$weight(c_j) = \sum_{w_i \in T} v_i \cdot k_j, \tag{2.14}$$

where w_i is an input text word, v_i is the weight of the word w_i and k_j is the weight of the concept c_j in the inverted index entry for w_i.

Milne and Witten. Milne and Witten (2008a) propose two relatedness measures for the Wikipedia knowledge base. The first measure is based on the Vector Space Model approach, where the relatedness of two Wikipedia articles is given by the cosine similarity between the article vectors. Rather than using $TF - IDF$ vectors based on term counts, the authors construct article vectors using link counts.

In this setting, each link is assigned a weight $w(s \rightarrow t)$, with s and t being the source and target articles respectively:

$$w(s \rightarrow t) = \log\left(\frac{|W|}{|T|}\right) \mid \text{if } s \in T,\ 0 \text{ otherwise}, \tag{2.15}$$

where W denotes all Wikipedia articles and T represents the set of all articles mentioning t.

The second measure is inspired by the *Normalized Google Distance* described in (Cilibrasi & Vitanyi, 2007), who propose a similarity measure using Google search engine results. Instead of search results, the authors in (Milne & Witten, 2008a) use the links present in Wikipedia articles to determine how related two articles are:

$$Relatedness_{MilneWitten}(a, b) = \frac{\log(\max(|A|, |B|)) - \log(|A \cap B|)}{\log(|W|) - \log(\min(|A|, |B|))}, \tag{2.16}$$

where a and b are two Wikipedia articles, A and B are the sets of all articles that link to a and b and W are all the articles in Wikipedia.

2.1.5 Hybrid Measures

Hybrid approaches to measuring the relatedness between concepts usually take advantage of the combination of multiple information sources. Li, Bandar, and McLean (2003) propose a nonlinear taxonomy-based model which incorporates shortest path and local density information in order to determine the similarity between words. In another line of research Tsatsaronis, Varlamis, and Vazirgiannis (2010) describe a text relatedness measure which combines the lexical similarity between two texts with the semantic relatedness computed for pairs of text words.

2.1.6 Ontology Quality

Several approaches have been proposed to analyze the properties of ontologies. Tartir, Arpinar, Moore, Sheth, and Aleman-Meza (2005) describe a methodology to evaluate the quality of an ontology, where one quality dimension considers the depth, breadth and height balance of the ontology inheritance tree. Burton-Jones, Storey, Sugumaran, and Ahluwalia (2005) propose a number of metrics based on semiotic theory to asses different aspects of ontology quality such as syntactic, semantic, pragmatic and social. Some of the aforementioned metrics have been adopted and extended to build ontology profiles for supporting the self-configuration of an ontology matching system (Cruz, Fabiani, Caimi, Stroe, & Palmonari, 2012).

Another line of research (Theoharis, Tzitzikas, Kotzinos, & Christophides, 2008) analyzes the graph features of Semantic Web schemas, more specifically power-law degree distributions. The authors note that the Semantic Web schemas which have a significant number of properties and/or classes (e.g. the Cyc ontology) approximate a power-law for total-degree distribution, where the total-degree represents the number of subsumed classes.

The structure of RDF graphs, e.g. the instantiated RDF classes of a resource or the properties, is leveraged to construct schemas of linked open data sources (Konrath, Gottron, Staab, & Scherp, 2012).

2.1.7 Comparison Between Existing Relatedness Measures

The measures described so far have a number of shortcomings. To start with, *concept definition based measures* require that every concept has associated a definition describing it. This definition is not present in all ontologies, and for all concepts. Moreover, concept definition-based measures which provide good results in the case of WordNet do not perform equally well when applied to other knowledge bases such as DBpedia or OpenCyc (Rusu et al., 2011). This is due to several reasons. Firstly, concepts in WordNet represent words and collocations in a lexicon: they have associated dictionary-like definitions and in some cases example sentences, whereas in OpenCyc, these definitions aid in describing the structure of the ontology. Secondly, two concepts that are similar do not necessarily have an overlap in their corresponding definitions.

Structure-based measures that rely on the distance between two concepts treat all edges uniformly. These measures work under the assumption that the distances between more specific concepts and the distances between more abstract concepts have the same interpretation. This, however, is not the case in most ontologies (Resnik, 1995).

The relatedness measures centered on the idea of semantically correct paths have been validated only in the case of WordNet. Also, Hirst and St-Onge's measure is specifically tailored to the relationships used in WordNet. Moreover, the direction of each relation is hard to determine for relations other than synonymy, antonymy, see also or taxonomic (Mazuel & Sabouret, 2008). Similarly to the distance-based measures, Hirst and St-Onge's measure treats all edges as being equally informative.

Information content-based measures do not have the disadvantages of the previously-mentioned structure-based measures, as the information content is independent of the distance between concepts or the depth of the concepts in the ontology (Pesquita et al., 2009). Yet they only take into account the information content of the two concepts and of their Least Common Subsumer for measuring similarity or relatedness.

The measures which estimate concept probabilities from word frequencies in a given corpus do not take word polysemy into account. Word frequencies and concept frequencies are not equivalent. For example, occurrences of the word "bus" cannot be uniquely mapped onto a single concept, but correspond to the following WordNet 3.0 concepts:

Bus_1 - a vehicle carrying many passengers,

Bus_2 - an electrical conductor that makes a common connection between several circuits.

An alternative to estimating concept frequencies from word frequencies is to use semantically-annotated corpora. However, acquiring these corpora is a time intensive and expensive process. Moreover, this process needs to be repeated whenever the domain changes as different application domains require different corpora.

The intrinsic and extended information content-based measures use the ontology itself as a statistical resource, and do not require additional semantically-annotated corpora for estimating concept probabilities.

2.2 Text Annotation

Annotating text with concepts defined in knowledge bases is equivalent to the *word sense disambiguation* task (Ide & Veronis, 1998). This task has seen three main approaches emerging along the years: *supervised*, *unsupervised* and *knowledge-based*. Supervised techniques employ machine learning methods for training a classifier on concept-labeled data; unsupervised methods rely on clustering of word contexts while knowledge-based approaches exploit various concept inventories like dictionaries, ontologies, thesauri to determine the appropriate concept for a given word in context.

By far the most popular source of annotations has been the WordNet lexical database. Throughout the years the Senseval and SemEval semantic evaluation workshops (Senseval, 2004; SemEval, 2012, 2013, 2014) provided datasets labeled with WordNet concepts, creating not only a common comparison setting for different annotation systems but also contributing with training data for supervised approaches. The best performing systems have been the supervised ones, although in recent semantic evaluation workshops (Agirre et al., 2010; Navigli, Jurgens, & Vannella, 2013) weakly supervised and knowledge-based techniques have been predominant. Due to its rich encyclopedic content, Wikipedia concepts were also deemed as valid annotation candidates, especially for named entities (Bunescu & Pasca, 2006; Cucerzan, 2007). In the bioinformatics domain the Gene Ontology was used as a controlled vocabulary (Andreopoulos, Alexopoulou, & Schröder, 2008). More recently, given the increased interest in multilingual applications, BabelNet (Navigli & Ponzetto, 2012a) was proposed as a multilingual concept inventory.

The remainder of this section describes related work from each of the three main approaches to text annotation and their application to different knowledge bases.

2.2.1 Supervised Approaches

Supervised approaches to text annotation use a variety of machine learning algorithms to learn a classifier based on manually labeled text. The training set comprises text fragments in which words or collocations are assigned concepts from a knowledge base. The features used for learning the model include words belonging to the local context, syntactic information such as part-of-speech or grammatical dependencies or semantic information such as named entities; the training data comprises datasets from evaluation workshops, parallel corpora or SemCor (Landes et al., 1998). SemCor was built from two textual corpora: a subset of the Brown corpus and Stephen Crane's novella *The Red Badge of Courage. More than half a million open-class words in the two corpora were semantically tagged using WordNet as a lexicon.*

One of the more simple classifiers used for text annotation is Naive Bayes (Duda, Hart, et al., 1973). Given a set of candidate concepts $C = \{c_1, c_2, ..., c_n\}$ for a word to annotate w and a set of context features $F = \{f_1, f_2, ..., f_m\}$ for w, this classifier learns the most appropriate candidate concept for the word to annotate as the concept maximizing the following probabilities:

$$\hat{c} = \underset{c_i \in C}{\operatorname{argmax}} P(c_i) \prod_{j=1}^{m} P(f_j|c_i) \tag{2.17}$$

The assumption is that the features are conditionally independent given the

concept and the probabilities are estimated based on relative frequency counts in the training corpus. Despite these drawbacks, text annotation systems using Naive Bayes classifiers (Leacock, Miller, & Chodorow, 1998) or ensembles of such classifiers (Pedersen, 2000) obtained competitive results on standard datasets (Y. K. Lee & Ng, 2002).

A popular machine learning algorithm for text annotation is Support Vector Machines (SVM) (Cortes & Vapnik, 1995). A Support Vector classifier learns separating hyperplanes which maximize the margin of the training data in a high dimension feature space. Chan, Ng, and Zhong (2007) propose an SVM-based approach trained on English-Chinese parallel corpora covering the most frequent nouns, adjectives and verbs in the Brown corpus, SemCor and the DSO corpus (Ng & H. B. Lee, 1996); this system achieved best results on the SemEval 2007 coarse-grained disambiguation task (Navigli, Litkowski, & Hargraves, 2007).

Other supervised approaches include maximum entropy classifiers (Novischi, Srikanth, & Bennett, 2007; Tratz et al., 2007) or perceptron-trained Hidden Markov Models (Ciaramita & Altun, 2006; Mihalcea, Csomai, & Ciaramita, 2007).

Even if these approaches generally outperform WordNet's most frequent sense baseline, which turns out to be hard to overcome, the main obstacle is the scarcity of sense-annotated corpora, especially as retraining is necessary for other domains or languages.

Mihalcea and Csomai (2007) coin the term *text wikification* as the task of linking unstructured text fragments to Wikipedia articles. The authors develop a system called *Wikify!* which performs keyword extraction and linking to the corresponding Wikipedia article. Two different algorithms are considered for linking: a) a knowledge-based technique inspired by the Lesk algorithm (Lesk, 1986) which determines the contextual overlap between the Wikipedia article and the paragraph where the word appears and b) a supervised Naive Bayes approach using local and topical features such as the part-of-speech of the word to annotate and of the context words.

Milne and Witten (2008b) propose a different supervised approach to wikification. They use Wikipedia both as a knowledge base for annotation and as a source of training data. As features the authors propose to balance the relatedness of a concept to the surrounding context and its prior probability. Their relatedness measure (Milne & Witten, 2008a) takes advantage of the Wikipedia link structure, while the prior probability of a concept is determined by the number of links pointing to this concept. A similar approach is proposed by Medelyan, Witten, and Milne (2008), however this algorithm considers all context terms as being equally relevant for annotation.

Document coherence was exploited in (S. Kulkarni, Singh, Ramakrishnan, & Chakrabarti, 2009) via collective optimization. The authors model the combination of *node potential* providing evidence of local coherence between the word to annotate and the Wikipedia candidate concept and *clique potential* indicating topical coherence of the concepts selected to annotate all words. Inference is solved heuristically using local hill-climbing and linear program relaxations.

Weakly supervised methods make use of seed concepts in order to guide the annotation process. This type of approach has had the best performance on domain-specific texts, where a small number of manually disambiguated concepts from the domain was used as seeds to improve the performance of the knowledge-based method (A. Kulkarni, Khapra, Sohoney, & Bhattacharyya, 2010).

2.2.2 Unsupervised Approaches

Unsupervised approaches perform *word sense induction* or *discrimination* by identifying the meaning of a word solely based on the corpus, which can be an unannotated monolingual one or parallel text. These methods usually involve clustering similar contexts of a word, where each cluster represents a different sense of that word.

The *context-group discrimination* algorithm proposed by Schütze (1998) represents words, contexts and senses in a high-dimensional space. Senses are obtained by clustering similar context vectors using a combination of the expectation maximization algorithm and agglomerative clustering. The author also investigates a different representation of context vectors via dimensionality reduction techniques such as singular value decomposition (Golub & Van Loan, 2012).

Lin and Pantel (2002) describe an alternative clustering algorithm called *clustering by committee*. In this case each word is represented as a vector encoding the pointwise mutual information between the word and its context; the similarity between two words is computed as the cosine of the angle between their corresponding pointwise mutual information vectors. The top k similar words are clustered using an average-link clustering approach, where the words in each cluster form a *committee*. New committees are created in an iterative manner provided they are not similar to the already-generated committees. In the discrimination step each word is assigned to its most similar cluster determined based on the similarity between the word pointwise mutual information vector and the committee centroid.

Graph-based approaches rely on building *co-occurrence graphs* from pairs of words which appear together in a given context. Veronis (2004) proposes the *HyperLex* algorithm which exploits the characteristics of small world graphs (Albert & Barabasi, 2002), i.e. most nodes can be reached from any other node via a small number of steps. A co-occurrence graph is built for each word w to be annotated; the graph nodes represent co-occurring words in the context of w, while the edges are weighted based on the relative frequency of the two words co-occurring.

Probabilistic models of text generation such as Latent Dirichlet Allocation (LDA) (Blei, Ng, & Jordan, 2003) have also been applied in a word sense induction setting. The LDA model represents each document as a mixture of K topics, with each topic being a distribution over words. Boyd-Graber, Blei, and Zhu (2007) extend the initial LDA model in order to identify document topics and senses for the words by modeling senses as a hidden variable. Instead of generating words from global topics, the work presented by Brody and Lapata (2009) describes a Bayesian framework which generates words from local topics using the local context of the word to annotate.

Evaluating unsupervised techniques which rely on clustering is quite challenging. Agirre and Soroa (2007) propose both a supervised and an unsupervised evaluation for word sense induction. The supervised evaluation is complementary to the standard (unsupervised) clustering evaluation technique, trying to overcome the bias towards a particular clustering approach.

2.2.3 Knowledge-based Approaches

Knowledge-based methods do not require labeled data and are easier to adapt to different domains or languages; the most important factor is the quality of the knowledge base. Ponzetto and Navigli (2010) show that a high quality knowledge

base enables straightforward definition-based and graph-based approaches to attain performances comparable to supervised techniques.

A simple definition-based approach is the Lesk algorithm (Lesk, 1986); the central idea is to determine the number of overlapping words between the definitions of candidate concepts (see Section 2.1.1). Two words in text are disambiguated by computing the similarity between each pair of concepts belonging to the set of candidate concepts of the two words (see Eq. 2.1) and selecting the concepts with the highest similarity. If $Concepts(w_1)$ and $Concepts(w_2)$ are the candidate concepts for the words w_1 and w_2, respectively, one would need to determine $|Concepts(w_1)|\cdot|Concepts(w_2)|$ definition overlaps in order to annotate the two words. Moreover, a context of n words would imply determining $\prod_{i=1}^{n}|Concepts(w_i)|$ overlaps. This lead to a simplified version of the algorithm where the overlap is determined between the definitions of candidate concepts and the words in the context. Banerjee and Pedersen (2002) propose an extension of the Lesk algorithm by considering not only candidate concept definitions but also definitions of related concepts such as hypernyms, hyponyms, etc.

Graph-based algorithms involve constructing a graph of concepts and relations between these concepts either by using the entire knowledge base (Agirre & Soroa, 2009) or a subset (Sinha & Mihalcea, 2007) and then applying ranking techniques to the concept graph in order to identify word annotations. The Personalized Page Rank algorithm (Agirre & Soroa, 2009) assigns the initial probability mass uniformly only to context nodes as opposed to the original PageRank algorithm (Brin & Page, 1998) where the probability mass is distributed uniformly to all graph nodes. Sinha and Mihalcea (2007) build a graph from the candidate concepts of a word and the concepts belonging to the word context. They use different similarity measures to determine the edges in the graph, and a number of centrality measures to rank the concepts. Structural Semantic Interconnections (Navigli & Velardi, 2005) is another graph-based approach which further develops lexical chains - sequences of semantically related words, proposed in (Morris & Hirst, 1991) - by encoding a context free grammar of valid semantic interconnection patterns. Navigli and Lapata (2010) compare different local and global graph connectivity measures for disambiguating words using WordNet as a sense inventory. Local measures such as Degree or Eigenvector centrality (including PageRank (Brin & Page, 1998) and HITS (Kleinberg, 1999)) quantify the relevance of a single node in the graph. Global measures such as Compactness, Graph Entropy or Edge Density take into account the graph structure as a whole. Their evaluation results show that local measures such as Degree and PageRank perform better than global measures.

DBpedia Spotlight (Mendes, Jakob, Garcia-Silva, & Bizer, 2011) is a tool for annotating text documents with DBpedia concepts. Their annotation approach is based on representing DBpedia resources using a Vector Space Model where each resource is weighted using a $TF-ICF$ weight similar to the $TF-IDF$ weight used in information retrieval. The difference between the two weighting schemes is that $TF-IDF$ is based on word frequencies at the document and corpus level whereas $TF-ICF$ determines the relevance of a word for a DBpedia resource or set of resources. More precisely, TF is the term frequency showing how relevant is a word for a given resource and ICF is the inverse candidate frequency, capturing the importance of a word given a set of candidate resources. Given this representation, the annotation task is seen as ranking the candidate concepts for a word to annotate based on the similarity score (cosine similarity) between the concept vectors and the

word context.

With the increase in popularity of the multilingual text annotation task, BabelNet (Navigli & Ponzetto, 2012a) was proposed as a concept inventory for the latest semantic evaluation workshop SemEval 2013 Task 12 (Navigli et al., 2013). The participating systems were required to provide either BabelNet, WordNet or Wikipedia annotations for the nouns in the test corpus. All three systems opted for a graph-based approach, either by a) constructing a graph of co-occurring lemmas in a ten sentence window around the word following the work of Navigli and Lapata (2010), b) identifying paths between the candidate concepts and the context based on an ant-colony algorithm (Schwab, Goulian, Tchechmedjiev, & Blanchon, 2012) or c) applying a Personalized Page Rank algorithm (Agirre & Soroa, 2009) extended with concept frequencies (Gutierrez Vazquez, 2012) on the graph obtained by expanding WordNet with domain information (Gutierrez Vazquez, Fernandez Orquin, Montoyo Guijarro, Vazquez Perez, et al., 2011). Only one system provided Wikipedia-based annotations. Aside from the systems participating in the SemEval workshop, Navigli and Ponzetto (2012b) harness BabelNet's multilingual knowledge base and propose a graph-based annotation approach which jointly exploits information about a concept available in multiple languages.

2.2.4 Comparison Between Existing Text Annotation Approaches

Each of the three text annotation approaches covered in Section 2.2.1, Section 2.2.2 and Section 2.2.3, respectively, have a number of advantages and disadvantages.

Supervised text annotation techniques have obtained the best results in semantic evaluation workshops, improving upon WordNet's most frequent sense baseline. The main drawback is the scarcity of training data as retraining is necessary for different domains or languages.

Unsupervised approaches, on the other hand, require no training or external knowledge bases and are easy to adapt to other domains or languages. The fact that these techniques are only based on unannotated monolingual corpora which are widely available or on parallel text makes them highly appealing. However, unsupervised techniques are harder to evaluate as words are not annotated with predefined concepts but rather the meaning of the word is induced from its context.

With the availability of machine-readable dictionaries, thesauri or ontologies spanning different domains *knowledge-based* approaches to text annotation have become increasingly popular. Such approaches exploit the information available in the knowledge base while requiring no training data. The quality of the knowledge base in terms of the concepts that it covers and types of relations between concepts plays an important role in the performance of these systems.

2.3 Our Contribution

The comparisons between existing relatedness measures (see Section 2.1.7) and existing text annotation approaches (see Section 2.2.4) show that each of the presented approaches has a number of disadvantages. In this thesis we describe a generic text annotation framework based on background knowledge and relying on concept relatedness, and aims to overcome some of these disadvantages by:

- proposing a concept definition-based measure of relatedness based on a Vector

Space Model which weights the contribution of relevant concept definitions instead of treating all definitions in a uniform manner;

- proposing a structure-based measure of relatedness based on a concept weighting scheme which allows to distinguish between the types of concepts which can appear in an ontology or knowledge base. Current approaches do not make the distinction between different types of concepts;
- combining the two types of relatedness measures in order to compensate for possible shortcomings of either the concept definition-based measure or the structure-based measure;
- defining an automatic text annotation framework which can be used to annotate words or collocations with concepts defined in different background knowledge datasets. Most of the annotation algorithms presented in the related work section have been developed having in mind a particular ontology or knowledge base. Moreover, we use a knowledge based approach to text annotation as this allows us to take advantage of the existing information available in knowledge bases and ontologies without the need for labeled data.

Chapter 3

The Proposed Relatedness Measures

The text annotation framework described in this thesis selects the most appropriate concept to annotate a word or collocation based on the relatedness between concepts belonging to the context of the word (collocation).

There are several aspects to take into account when determining the relatedness between concepts represented in ontologies or knowledge bases. Firstly, ontologies or knowledge bases are structured in different ways, depending on the purpose for which they are built. Cyc, for example, is based on a cross-domain ontology which has a number of abstract concepts grouping information. WordNet, on the other hand, is a lexical database where the concepts represent words and collocations. If the relatedness measure relies on determining the distance between two concepts, an important requirement is that concept distances can be interpreted in a consistent manner (Pirro & Euzenat, 2010). In the case of information content-based measures, more abstract concepts have higher probability of occurrence, hence less information content. The information content corresponding to the unique top concept of an ontology is zero (Resnik, 1995). Secondly, the way conceptualizations are specified via ontology classes, instances, object properties, etc. is not consistent across ontologies (see Section 1.1). The problem arises when determining the concept distance – i.e. the number of semantic connections – between a class and an object property. Thirdly, some ontologies provide additional information for concepts, like a description of the concept, or various examples containing the concept. In WordNet, each concept has a succinct definition, a list of synonyms and in some cases an example sentence. The purpose of the concept descriptions can vary from one ontology to another; in WordNet the descriptions are similar to dictionary entries, in Cyc descriptions are meant as documentation for the ontology engineer and in DBpedia descriptions are written like encyclopedia entries. As a consequence of these differences, similarity measures that are solely based on concept definitions can provide poor results (Rusu et al., 2011).

This chapter proposes measures of relatedness between concepts which use a) the concept definitions b) the knowledge base structure and c) a hybrid approach which combines the aforementioned measures. The structure-based relatedness measure was described in (Rusu, Fortuna, & Mladenić, 2014).

3.1 Definition-based Concept Relatedness

The *Vector Space Model* has been a very popular model used to represent documents in information retrieval. Schütze (1998) proposed an unsupervised word sense dis-

ambiguation algorithm based on clustering where words, contexts and senses are represented using the Vector Space Model. The *Word Vectors* are obtained for each word w by counting words co-occurring with w within a given window such as a sentence or a paragraph. *Context Vectors* are represented by the centroid of the word vectors which belong to the context. *Sense Vectors* are clusters of all context vectors identified for an ambiguous word in the corpus. Following Schütze's work, Patwardhan (2003) introduces a measure of semantic relatedness which relies on context vectors. In his approach, each WordNet concept is represented as a *Definition Vector*, obtained by computing the centroid of the word vectors which appear in the concept definition. The relatedness between two concepts is defined as the cosine similarity between the corresponding definition vectors.

In this work we propose an extension of *Definition Vectors* based on a kernel function which leverages the contribution of different concept definitions.

3.1.1 Extended Definition Vectors

The *Extended Definition Vectors* measure adapts a web-based kernel function for measuring the relatedness of short text snippets defined in (Sahami & Heilman, 2006). This kernel function determines the relatedness between two text snippets by considering web search engine results obtained when using the snippet as a query. The returned documents representing a *context vector* for the initial text snippet are compared with a cosine measure in order to determine the relatedness of the snippets. In our case the two text snippets are concepts from a knowledge base, while the context vector of a concept is composed of the definition of the concept and definitions of connected concepts. Depending on the ontology or knowledge base, we take into account different connected concepts. In the case of WordNet we show evaluation results when a) the connected concepts are related only via taxonomic relations and b) the connected concepts are related via any type of relation (see Table 6.2). For OpenCyc and DBpedia we report results when using the relations listed in Section 4.4; we refer the reader to Table 6.3 and Table 6.5 for the OpenCyc and DBpedia evaluation results, respectively.

We represent the knowledge base as a graph $G = (V, E)$ where V is the set of all concepts in the knowledge base, and E represents the relationships between these concepts. In this representation, each node v corresponding to a concept has assigned a definition d_v describing the node. Example definitions are the concept gloss in WordNet or the resource abstract in DBpedia (see Chapter 4). Let $S(v) = \{d_{v_i}\} \forall v_i \in V | Path(v, v_i) \leq m$ be the set of definitions associated to nodes related to v, where each of these nodes is connected to v via a path of length at most m; note that $d_v \in S(v)$.

The concept relatedness algorithm based on extended definition vectors for computing the relatedness between two concepts represented by two nodes in the graph v and w is described in Algorithm 3.1.

As a first step, we compute the $TF - IDF$ term vector t_i for each definition $d_{v_i} \in S(v)$. Next we determine the centroid $C(v)$ of the L_2 normalized vectors t_i:

$$C(v) = \frac{\sum_{i=1}^{n} \alpha_i \frac{t_i}{\|t_i\|_2}}{\sum_{i=1}^{n} \alpha_i}, \tag{3.1}$$

where $n = |S(v)|$ and α_i is a weight associated with each term vector. The intuition is that term vectors should not be equally relevant for determining the centroid. The

Algorithm 3.1: The concept relatedness algorithm based on extended definition vectors.

Data: $G(V, E)$
v, w two nodes in the graph
$S(v)$, $S(w)$ the sets of definitions for nodes v and w
$\alpha = \alpha_1 + \alpha_2 + ... + \alpha_n$ weights associated with node definitions in $S(v)$
$\beta = \beta_1 + \beta_2 + ... + \beta_k$ weights associated with node definitions in $S(w)$
Result: the relatedness between v and w

`/* the term vectors for the definitions in` $S(v)$ `and` $S(w)$ `*/`
T_v = term vectors for all definitions in $S(v)$
T_w = term vectors for all definitions in $S(w)$
`/* the centroid for` v `*/`
$SC(v) = 0$
for *each term vector in* T_v **do**
  $SC(v) = SC(v) + \alpha_i \frac{t_{v_i}}{\| t_{v_i} \|_2}$
end
$C(v) = \frac{1}{\alpha} SC(v)$
`/* the centroid for` w `*/`
$SC(w) = 0$
for *each term vector in* T_w **do**
  $SC(w) = SC(w) + \beta_i \frac{t_{w_i}}{\| t_{w_i} \|_2}$
end
$C(w) = \frac{1}{\beta} SC(w)$
`/* the extended definition vector for` v `*/`
$ED(v) = \frac{C(v)}{\| C(v) \|_2}$
`/* the extended definition vector for` w `*/`
$ED(w) = \frac{C(w)}{\| C(w) \|_2}$
`/* the relatedness kernel between` v `and` w `*/`
$K(v, w) = ED(v) \cdot ED(w)$

definition of the node v is the most relevant compared to definitions of connected nodes, and the corresponding term vector should therefore have the highest weight. Moreover, the term vector weight of a node v_i should be inversely proportional to the length of the path between v and v_i. In the evaluation settings (see Chapter 6) we set the weight of the term vector corresponding to the node v to 1 and experiment with different values between 0 and 1 for the weight of the term vector corresponding to a connected node v_i.

Next, we define the *extended definition vector* $ED(v)$ as the L_2 normalization of the centroid $C(v)$:

$$ED(v) = \frac{C(v)}{\| C(v) \|_2} \tag{3.2}$$

Given two nodes v and w, the relatedness kernel is defined as:

$$K(v, w) = ED(v) \cdot ED(w) \tag{3.3}$$

Figure 3.1 shows the graphical interpretation of the relatedness kernel K.

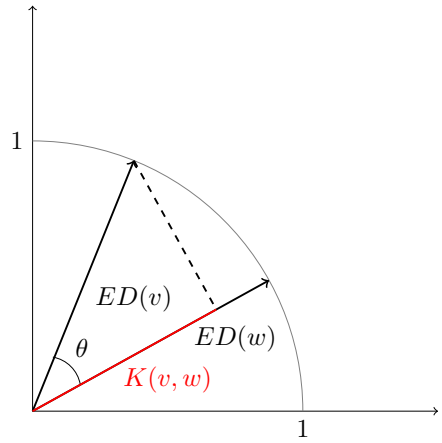


Figure 3.1: The relatedness kernel $K(v, w)$ defined as the cosine between the extended definition vectors $ED(v)$ and $ED(w)$.

The main difference between the *Definition Vectors* proposed in (Patwardhan, 2003) and the *Extended Definition Vectors* that we propose in this work lies in the way we obtain the vectors from concept definitions. In the case of *Definition Vectors*, the concept definition of a concept c is augmented with that of connected concepts which are directly related to c. Moreover, all connected concept definitions are treated as being equally important for determining the relatedness score. The drawback is that we cannot extend the set of connected concept definitions without treating all definitions as being equally relevant. Additionally, we cannot differentiate between connected concept definitions based on the type of relation. To overcome these drawbacks we propose a more general method (*Extended Definition Vectors*) which takes into account the (weighted) contribution of each concept definition. The concept definition weight is a parameter which is estimated based on a validation dataset (see Section 6.1.3 and Section 6.1.5).

Figure 3.2 graphically depicts different approaches to constructing vectors from concept definitions. In this figure we suppose a simple scenario: there is a concept represented by a node v in the knowledge base graph; suppose v has associated two definitions for which we compute the $TF - IDF$ term vectors t_1 and t_2. Figure 3.2a describes the case when the two definitions are merged yielding one (longer) definition having the term vector t; the definition vector is therefore the normalized term vector of t. This is the approach described in Patwardhan (2003). Figure 3.2b describes how we obtain the extended definition vector by summing the normalized unweighted term vectors of t_1 and t_2. In Figure 3.2c we also associate a weight with each term vector.

3.2 Structure-based Concept Relatedness

Our approach is based on the geometric model described in cognitive psychology, and inspired from Rada et al. (1989) work on defining a distance metric on semantic

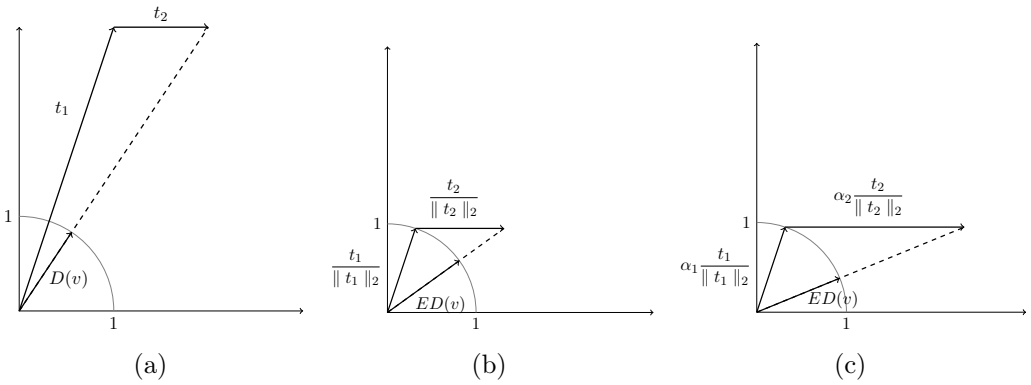


Figure 3.2: Different approaches to constructing vectors from concept definitions. (a) shows the *Definition Vector* $D(v)$ for a node v where the two definitions associated with v have been merged. (b) shows the *Extended Definition Vector* $ED(v)$ for a node v given two unweighted term vectors corresponding to two definitions associated with v. (c) shows the *Extended Definition Vector* $ED(v)$ for a node v given two weighted term vectors corresponding to two definitions associated with v; note that $\alpha_2 > \alpha_1$.

nets. We can view taxonomies such as MeSH, lexical databases such as WordNet or general-purpose ontologies such as OpenCyc as a *semantic network* where the nodes are the concepts and the links represent relationships between concepts.

In this work, we propose an extension of the distance metric which is based on assigning weights to knowledge base concepts and aggregating these weights in an effective manner. Concepts can be distinguished based on their degree of abstractness. **More abstract (or general) concepts** have a higher number of relations, where by *relation* we understand any relation between two concepts. Section 4.4 lists the relations that we consider for each Linked Dataset. In OpenCyc, for example, the concept *NaturalThing* has more than 100 taxonomic relations to other concepts; some of the concepts, such as *NaturalFeatureType* with more than 200 such relations, are used for meta-modeling. **More specific concepts** have a lower number of relations and are useful when solving tasks such as automatic text annotation. For example, the concept *Forest* in OpenCyc has slightly more than 30 relations; the WordNet concepts *coast* or *shore* each have a few more than 20 relations.

Throughout our experimental evaluation we show that by differentiating between concept types rather than considering all concepts in a uniform manner we can improve the results of the basic distance metric.

We consider the knowledge base as a graph $G = (V, E)$ where V is the set of all concepts in the knowledge base, and E represents a set of relationships between these concepts. The extension which we propose relies on three observations:

Observation 1 - Concept weights. A weight can be assigned to concepts in order to facilitate distinguishing between abstract and specific concepts. We propose to use the degree of a node representing a concept as the concept weight, having in mind that more abstract nodes usually have higher node degrees.

Observation 2 - Relation weights. The weight of an edge representing a relation can be defined as a function of its two adjacent nodes (concepts), penalizing edges where at least one of the nodes represents an abstract concept with a higher

number of relations.

Observation 3 - Concept relatedness. The relatedness between two concepts can be determined based on their weighted shortest path.

In the following sub-sections we define the concept and relation weights and present an algorithm for computing the weighted shortest path between two given concepts.

3.2.1 Concept Weights

Given the knowledge base represented as a graph $G = (V, E)$, the goal is to define a weight associated to each graph node, which would enable distinguishing between node types.

Inspired by previous work on node and edge weighting schemes (see Section 2.1.2), we study the applicability of using node degrees as a weight assigned to the graph nodes. The degree of a node is defined as the sum of in-links and out-links of that node. To construct a reasonable weight on the basis of node degrees, we apply a suitable transformation. We have experimented with two such functions – the logarithm and the square root.

$$\begin{aligned} CW : V &\rightarrow (0, log(Deg_{max})] \\ CW(v) &= log(Degree(v)), \end{aligned} \tag{3.4}$$

and

$$\begin{aligned} CW : V &\rightarrow [0, \sqrt{Deg_{max}}] \\ CW(v) &= \sqrt{Degree(v)}, \end{aligned} \tag{3.5}$$

where Deg_{max} is the maximum degree of nodes in V and $Degree(v)$ is the degree of a node $v \in V$ defined as the sum of in-links and out-links of that node. If the degree of a node is 0, meaning the node has no relations, we assign that node degree a small value $\varepsilon < 1$.

3.2.2 Relation Weights

As noted in Observation 2, we combine the weights of adjacent nodes to obtain the edge weight. For the corpora that we used in the evaluation settings (see Section 6.1), we have conducted an empirical comparison in order to determine a suitable function for combining node weights into a weight of the corresponding edge. This comparison indicates that the maximum function is appropriate for penalizing edges with at least one adjacent node of high degree. Once the edge weight is calculated, the second step of our approach comprises the aggregation of edge weights, thereby determining the (weighted) shortest path between two concepts.

We define the RW as the weight assigned to each relation between two concepts:

$$\begin{aligned} RW : E &\rightarrow F \\ RW(v_i, v_j) &= \max(CW(v_i), CW(v_j)), \\ \forall \text{ edge } (v_i, v_j) \in E \text{ and } F &\subset (0, log(Deg_{max})] \text{ or } F \subset [0, \sqrt{Deg_{max}}], \end{aligned} \tag{3.6}$$

where Deg_{max} is the maximum degree of nodes in V.

Algorithm 3.2: The concept distance algorithm based on shortest weighted paths in a graph.

Data: $G(V, E)$
Result: pairwise distances for the graph nodes

```
/* determine the concept weight using one of the two defined concept
   weights; here we use the logarithm of the node degree            */
```
for *each node in* V **do**
  $CW(v) = log(Degree(v))$
end
```
/* determine the relation weight                                    */
```
for *each edge* $(v_i, v_j) \in E$ **do**
  $RW(v_i, v_j) = \max(CW(v_i), CW(v_j))$
end
```
/* determine the pairwise distance between two nodes by computing the
   shortest weighted path; keep the maximum distance                */
```
$Dist_{max} = 0$
for *each pair of nodes* $v_i, v_j \in V \times V$ **do**
  $DS(v_i, v_j) = ShortestWeightedPath(v_i, v_j)$
  if $DS(v_i, v_j) > Dist_{max}$ **then**
    $Dist_{max} = DS(v_i, v_j)$
  end
end

3.2.3 The Concept Relatedness Algorithm

Having decided on the concept and relation weights, the next step is to apply them for determining the similarity between concepts. As most graph algorithms take into account edge weights instead of node weights, we consider the previously defined edge weights, where an edge represents a relation between two concepts defined in the knowledge base. Similar to Rada et al.'s work, the conceptual distance represented by the shortest path between two concepts is a decreasing function of relatedness, i.e. the smaller the conceptual distance is, the more related the concepts are.

Algorithm 3.3: The concept relatedness algorithm based on the concept distance.

Data: $G(V, E)$
Result: pairwise relatedness for the graph nodes

```
/* determine the pairwise relatedness between two nodes based on the
   distance between the nodes                                       */
```
for *each pair of nodes* $v_i, v_j \in V \times V$ **do**
  $NDS(v_i, v_j) = \dfrac{DS(v_i, v_j)}{Dist_{max}}$
  $R(v_i, v_j) = 1 - NDS(v_i, v_j)$
end

The algorithm for computing the distance between two concepts represented by two nodes in the graph v_i and v_j, using weighted concept paths, is described in Algorithm 3.2. We start by determining the weight of each node; in Algorithm 3.2

we weighted each node using the logarithm of its degree (see line 2). Next, the weight of each edge is found in line 5. Finally, using these edge weights, we apply the shortest path algorithm (e.g. Dijkstra) for each pair of nodes in line 9.

The distance between two concepts is defined as:

$$DS : V \times V \rightarrow Y$$
$$DS(v_i, v_j) = ShortestWeightedPath(v_i, v_j), \tag{3.7}$$

where $V \times V$ is the Cartesian product of the set of concepts with itself, $Y \subset [0, Dist_{max}]$ and $Dist_{max}$ represents the maximum distance between pairs of nodes in V (see Eq. 3.8).

$$Dist_{max} = \max(DS(v_i, v_j)), \forall\ v_i, v_j \in V \times V \tag{3.8}$$

The distance between two identical concepts is zero.

$$DS(v_i, v_i) = 0 \tag{3.9}$$

Dijkstra's graph search algorithm determines the shortest path between two nodes in a graph having non-negative edge weights. Starting from a source node, the algorithm gradually constructs the paths with lowest weight from the initial node to all other neighbors.

In order to calculate correlations with human judgments of relatedness, we transform the distance measure into a relatedness measure. The distance obtained by applying Eq. 3.7 is normalized as follows:

$$NDS : V \times V \rightarrow Y_N$$
$$NDS(v_i, v_j) = \frac{DS(v_i, v_j)}{Dist_{max}}, \tag{3.10}$$

where $Y_N \subset [0, 1]$. The normalized conceptual distance is a decreasing function of relatedness, as shown in Algorithm 3.3, line 3.

3.3 Hybrid Approach

The hybrid approach proposed in this work weights the contribution of the definition-based relatedness and the structure-based relatedness between two concepts represented as the nodes v and w in the graph G:

$$H(v, w) = \zeta K(v, w) + (1 - \zeta)R(v, w), \tag{3.11}$$

where ζ is the *hybrid weight*.

The ζ parameter weights the contribution of the definition-based measure and the structure-based measure respectively for computing the final relatedness result. $\zeta = 0$ for knowledge bases where concepts have no associated definitions; alternatively, $\zeta = 1$ for knowledge bases which consist of a list of concepts and their definitions (e.g. dictionaries).

3.4 Summary

This chapter proposed three concept relatedness measures relying on concept definitions (Section 3.1), on the ontology or knowledge base structure (Section 3.2) and a hybrid approach which is a combination of the two (Section 3.3).

The automatic text annotation framework which integrates these relatedness measures and links words or collocations in text with concepts defined in background knowledge datasets is presented in Chapter 5. The following chapter (Chapter 4) presents the background knowledge datasets in more detail, before describing the actual annotation framework.

Chapter 4

Linked Datasets as Background Knowledge

Linked Open Data (LOD) currently contains over 62 billion triples from more than 900 datasets [1] spanning domains such as media, geography, publications, life sciences, etc., incorporating several cross-domain datasets. This important source of structured data has been used for building a variety of applications such as Linked Data browsers or search-engines as well as domain-specific applications such as semantic tagging and rating (Bizer, Heath, & Berners-Lee, 2009). A recent initiative is the development of the *Linguistic Linked Open Data (LLOD)* dedicated to linguistic resources (Chiarcos, Hellmann, & Nordhoff, 2012). The Linguistic Linked Open Data includes different LOD datasets grouped in three main categories:

- *lexical-semantic datasets* such as DBpedia, OpenCyc, Yago or WordNet,
- *digital libraries* such as Gutenberg, Open-Library or Rosetta-Project,
- *annotated corpora* such as Alpino-RDF.

In this chapter we present three of the main lexical-semantic datasets that are part of the Linguistic Linked Open Data, namely WordNet, OpenCyc and DBpedia. WordNet is a lexical database for English and the concept inventory of choice for the text annotation task in numerous semantic evaluation workshops. OpenCyc is the open source version of Cyc, a common-sense knowledge base primarily developed for modeling and reasoning about the world. The DBpedia knowledge base was created by extracting structured information from Wikipedia, a collaboratively edited encyclopedia. We choose these three linked datasets as background knowledge for our text annotation framework as they are all cross-domain datasets and have a broad coverage. For each of these datasets we present their main characteristics and an illustrative example. WordNet, OpenCyc and DBpedia are represented as a graph where the nodes constitute the concepts and the edges are the relations between these concepts. As the structured relatedness measure defined in Section 3.2 relies of node degrees, we also show, for each dataset, the distribution of node degrees. The *node degree* is obtained by counting the edges which are incident to that node.

[1] http://stats.lod2.eu/ Accessed April 2014

4.1 WordNet

WordNet (Fellbaum, 2005) is a lexical database for English; similar databases exist for other languages (Open Multilingual WordNet, 2014). Van Assem et al. (2006) present a standard conversion of WordNet to RDF/OWL. In this representation, the WordNet schema is based on three main classes: *Synset*, *WordSense* and *Word*. The *Synset* and *WordSense* classes have subclasses corresponding to four parts of speech: nouns, adjectives, verbs and adverbs. The *Word* class has as subclass *Collocation*. Moreover, each instance of Synset, WordSense and Word classes has associated a corresponding URI.

Table 4.1: The WordNet 3.0 synsets associated with the word senses *bus, autobus, coach, etc.* and the word senses *busbar, bus*, respectively.

Example noun synset	*bus, autobus, coach, charabanc, double-decker, jitney, motorbus, motorcoach, omnibus, passenger vehicle* (a vehicle carrying many passengers; used for public transport) "he always rode the bus to work"
URI	`http://purl.org/vocabularies/princeton/wn30/synset-bus-noun-1`
Word senses	*bus* `http://purl.org/vocabularies/princeton/wn30/wordsense-bus-noun-1`, *autobus* `http://purl.org/vocabularies/princeton/wn30/wordsense-autobus-noun-1`, *coach* `http://purl.org/vocabularies/princeton/wn30/wordsense-coach-noun-5`, etc.
Gloss	a vehicle carrying many passengers; used for public transport
Examples	he always rode the bus to work
Example noun synset	*busbar, bus* (an electrical conductor that makes a common connection between several circuits) "the busbar in this computer can transmit data either way between any two components of the system"
URI	`http://purl.org/vocabularies/princeton/wn30/synset-busbar-noun-1`
Word senses	*busbar* `http://purl.org/vocabularies/princeton/wn30/wordsense-busbar-noun-1`, *bus* `http://purl.org/vocabularies/princeton/wn30/wordsense-bus-noun-3`
Gloss	an electrical conductor that makes a common connection between several circuits
Examples	the busbar in this computer can transmit data either way between any two components of the system

A *synset* groups one or more synonyms. For example $bus_1 = \{bus, autobus, coach\}$ and $bus_2 = \{busbar, bus\}$ are two synsets which both contain the literal "bus", but which have different meanings: the first synset is defined

as "a vehicle carrying many passengers [...]", while the second synset is defined as "an electrical conductor [...]". In the WordNet datamodel "a *synset* contains one or more *word senses* and each word sense belongs to exactly one synset. In turn, each word sense has exactly one *word* that represents it lexically, and one word can be related to one or more word senses." (Van Assem et al., 2006).

A synset has the following characteristics: a) a corresponding URI, b) one or more *word senses*, c) a *gloss*, which is a brief definition of the synset, d) example sentences showing the usage of the synset members in text and e) relations to other synsets. Table 4.1 exemplifies each of these characteristics for the synset associated with the word senses *bus, autobus, coach, etc.* and the word senses *busbar, bus.*

4.1.1 Linked Dataset Overview

Table 4.2 gives an overview of the WordNet 3.0 lexical database. The RDF/OWL representation of WordNet includes ten relations defined between synsets (hyponymy, entailment, similarity, member meronymy, substance meronymy, part meronymy, classification, cause, verb grouping, attribute), and five between word senses (derivational relatedness, antonymy, see also, participle, pertains to). In this work we consider the entire synset as a concept which can be used to annotate words in context and we take into account the relationships between synsets.

Table 4.2: An overview of the WordNet 3.0 English lexical database.

WordNet 3.0	
Synsets	117,659
Noun synsets	82,115
Verb synsets	13,767
Adjective synsets	18,156
Adverb synsets	3,621
Relations between synsets	290,481
Word senses	206,941
Noun word senses	146,312
Verb word senses	25,047
Adjective word senses	30,002
Adverb word senses	5,580
Relations between word senses	87,111

We represent WordNet as a graph $G_W = (V_W, E_W)$, where V_W is the set of all nodes which constitute synsets and E_W denotes all the edges which are the relations between the synsets. Figure 4.1 shows the distribution of node degrees in WordNet 3.0. This lexical database is mainly built around hierarchical relationships, e.g. hypernym-hyponym, with most nodes having an even degree due to relation symmetry.

The node with the highest degree of about 1,300 represents the synset:

city, metropolis, urban center - a large and densely populated urban area; may include several independent administrative districts.

The concept *city* has a high number of *instance* relations to specific city names such as *New York*.

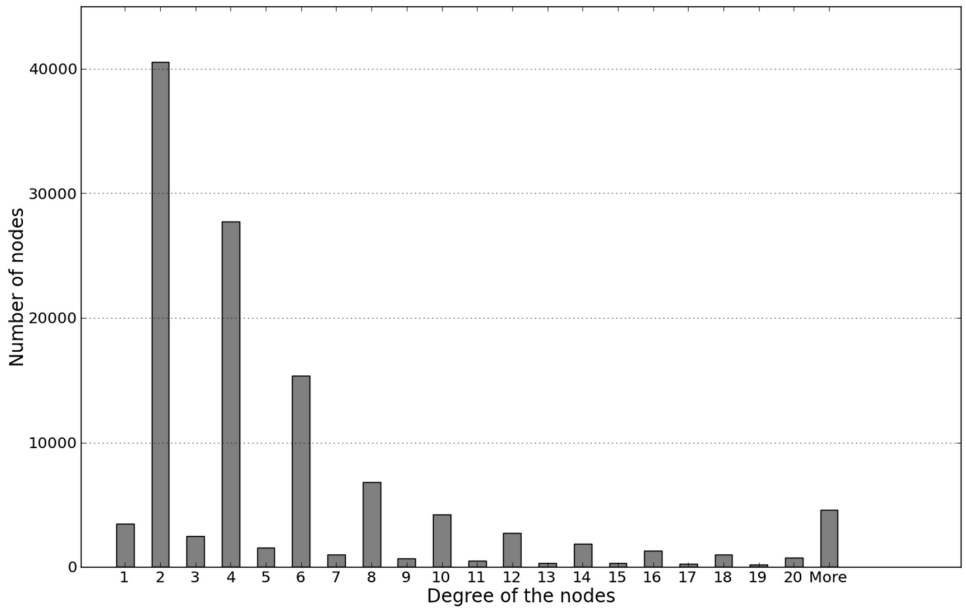


Figure 4.1: The distribution of node degrees in WordNet 3.0.

Nodes of degrees two, four or six account for more than 70% of the concepts. However, about 4% of the nodes have degrees above 20.

4.1.2 Illustrative Example

Figure 4.2 exemplifies five concepts described by WordNet synsets and the relationships between them. The synset having as word senses *bus, autobus, coach, etc.* can serve as annotation for the words "coach" or "bus". Similarly, the synset with word senses *busbar, bus* can be used to annotate the words "bus" or "busbar".

4.2 OpenCyc

OpenCyc (OpenCyc, 2014) is the open source version of the common-sense knowledge base Cyc (Lenat, 1995), covering about 40% of the complete Cyc knowledge base. It is also available as a downloadable OWL ontology. In this thesis we refer to the 15-08-2010 version of OpenCyc. The OpenCyc OWL ontology includes descriptions of classes, properties (mainly object properties) and instances. There are several types of relationships in OpenCyc, e.g. *rdf:type* is defined as a relation between an instance and a class, *rdfs:subClassOf* as a relation between a more specific class and a more general class. The OWL classes represent the most basic concepts in a domain, while the OWL object properties represent relations between instances of two classes. For example, the object property *friends*, with the domain and range *SentientAnimal*, relates instances of the class *SentientAnimal*. Table 4.3 shows the information associated with the OpenCyc concepts *Bus-RoadVehicle* and *ComputerBus*. The *cycAnnot:label* property denotes an OpenCyc concept identifier, while *rdfs:label* and *prettyString* are the concept natural language identifiers

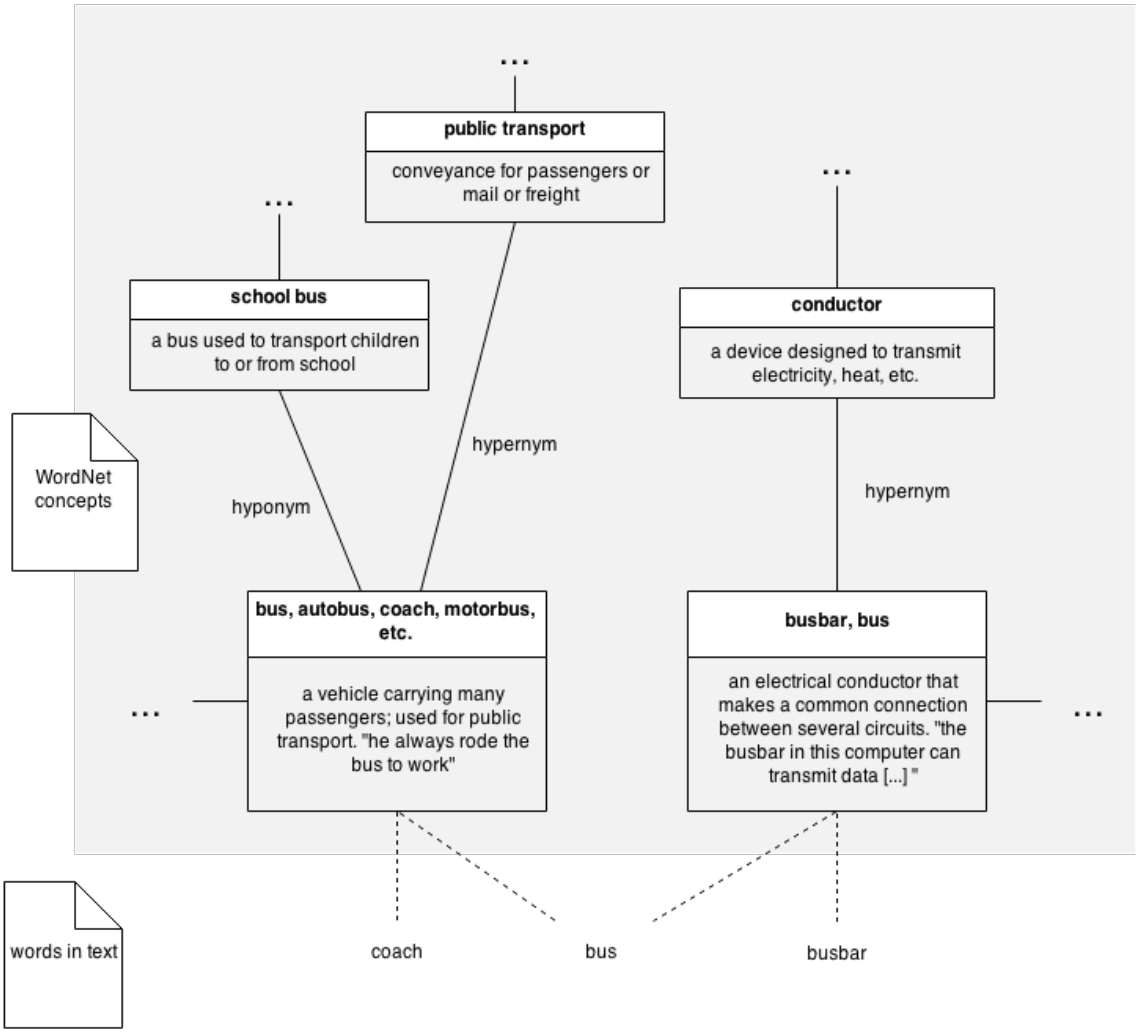


Figure 4.2: Five concepts described by WordNet synsets and the relationships between them. The synset having as word senses *busbar, bus* is a valid annotation for the words *busbar* or *bus*.

(NLIs) providing a human-readable version of the concept. The concept definition is represented via the *rdfs:comment* predicate.

4.2.1 Linked Dataset Overview

There are about 160,000 concepts (classes and instances) and nearly 16,000 object properties defined in this version of OpenCyc, describing more than 375,000 English terms. Roughly 65,000 of the concepts and object properties have an associated description. Table 4.4 lists a more detailed count of the concepts and a subset of the relationships between them, as obtained from the OWL version of OpenCyc. In the case of relationships, we consider the ones most common in the ontology. These are relationships between instances and classes, between classes and super-classes, and *broaderTerm*, a Cyc-specific relation. *BroaderTerm* indicates relations between concepts that are not strictly taxonomic.

We represent OpenCyc as a graph $G_O = (V_O, E_O)$, where V_O is the set of all OpenCyc concepts represented via classes, instances and object properties and E_O

Table 4.3: The OpenCyc concepts associated with the word *bus*.

Example concept	*Bus-RoadVehicle*
	[...] a ground transportation vehicle designed to carry many passengers [...]
cycAnnot:label	Bus-RoadVehicle
rdfs:label	bus
prettyString	bus, autobus, omnibus, [...]
rdfs:comment	[...] a ground transportation vehicle designed to carry many passengers [...]
Example concept	*ComputerBus*
	[...] a device which transmits data from one part of the Computer to another. [...]
cycAnnot:label	ComputerBus
rdfs:label	computer bus
prettyString	bus, buses, busses, computer buses, computer busses
rdfs:comment	[...] a device which transmits data from one part of the Computer to another. [...]

denotes all the relations between the concepts: *rdf:type*, *rdf:subClassOf* and *broaderTerm*. Figure 4.3 shows the distribution of node degrees in OpenCyc. In this case, about 59% of the nodes have degrees 1 or 2, while slightly less than 2% of the nodes have degrees more than 20. Moreover, we observe that abstract nodes have higher node degrees than more specific ones. For example, the concepts *ExistingObjectType* and *SpatiallyDisjointObjectType* have node degrees above 10,000, while concepts like *Boat* or *Canoe* have node degrees of 20 and 6, respectively.

Table 4.4: OpenCyc OWL 15-08-2010 Version concepts and a subset of relationships between concepts.

OpenCyc OWL 15-08-2010 Version	
OWL classes	69,994
Instances	91,287
Relations between an instance and a class	178,150
Relations between a class and a superclass	112,556
CYC broaderTerm	132,607

4.2.2 Illustrative Example

Figure 4.4 exemplifies different OpenCyc concepts and the relationships between them. Both the *Bus-RoadVehicle* and *ComputerBus* concepts can serve as annotations for the word "bus". *Bus-RoadVehicle* is an instance of *VehicleTypeByIntendedUse* and has the concept *PublicTransportationDevice* as a broader term. *ComputerBus* is a subclass of *ComputerHardwareComponent*.

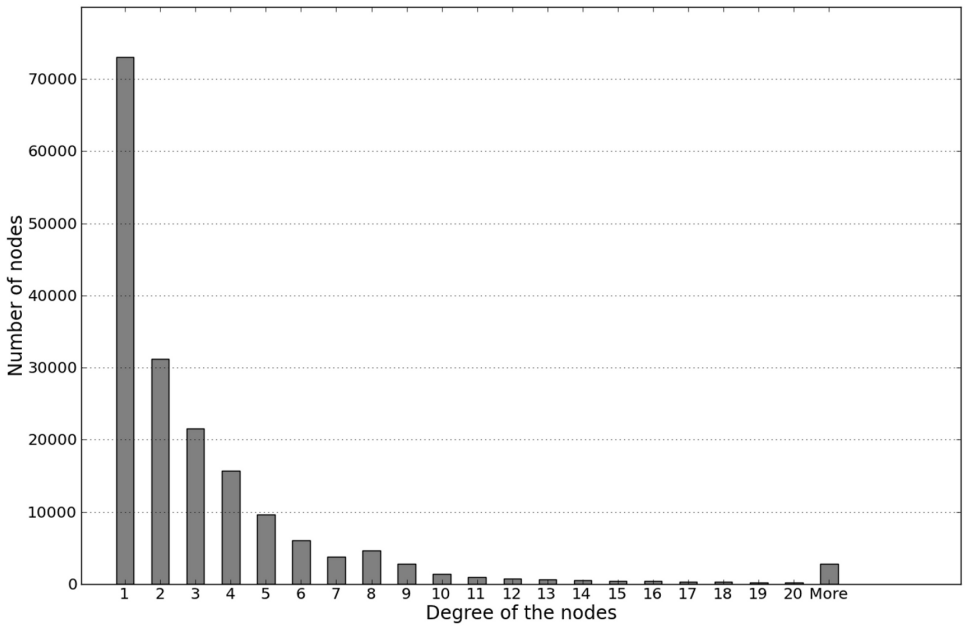


Figure 4.3: The distribution of node degrees in OpenCyc.

4.3 DBpedia

DBpedia (Lehmann et al., 2014) is a project aimed at extracting structured information from Wikipedia infoboxes. The results is a multilingual knowledge base currently including 119 languages. Due to the fact that it is a general knowledge base covering a variety of topics many datasets published as Linked Data have RDF links pointing to DBpedia, making it a "central interlinking hub" for Linked Open Data (Lehmann et al., 2014).

The main building block of the DBpedia knowledge base is the *resource* having a URI-based reference of the form `http://dbpedia.org/resource/Name` derived from the corresponding Wikipedia article URL `http://en.wikipedia.org/wiki/Name`. Each resource is associated with a *label*, a long (maximum 3000 characters) and short (maximum 500 characters) *abstract* obtained from the Wikipedia page text content and a *link to the Wikipedia page*. Table 4.5 shows example DBpedia concepts represented by the DBpedia resources `http://dbpedia.org/resource/Bus` and `http://dbpedia.org/resource/Bus_(computing)`. The short and long abstracts are denoted by the *rdfs:comment* predicate and *dbpedia-owl:abstract* predicate, respectively. We also show a subset of categories for this concept, represented by the *dcterms:subject* predicate. This resource is an instance of the *Municipality* class (see the *rdf:type* predicate).

4.3.1 Linked Dataset Overview

The latest version of the project (3.9) includes a knowledge base of approximatively 4 million resources, 3.2 million of them being classified into a shallow ontology spanning across multiple domains. This ontology includes around 500 classes, mainly

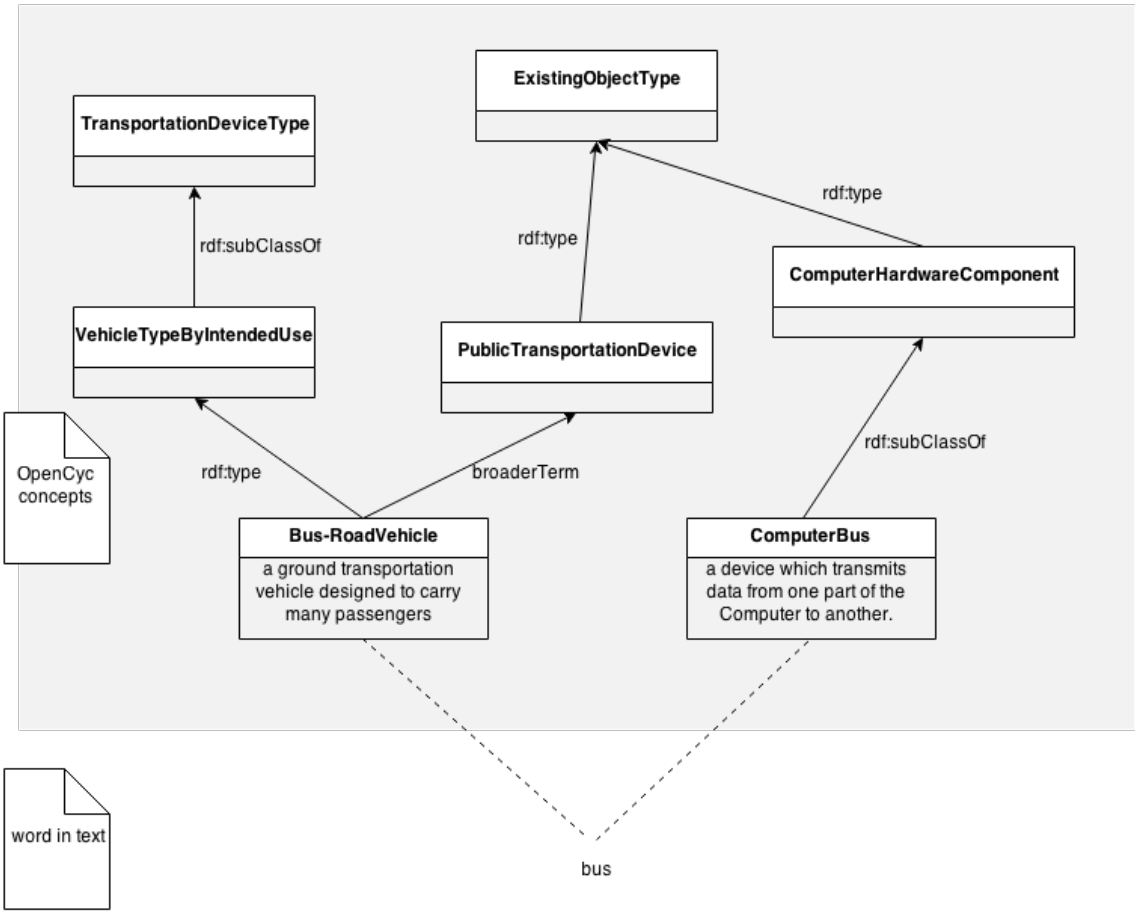


Figure 4.4: Example OpenCyc concepts and relations between concepts.

representing *places, persons, species, organizations* or *creative works* (e.g musical work, films, etc.). The classes are organized in a subsumption hierarchy, have more than 2,000 properties. In this work we are using a slightly older version of the ontology, namely 3.2. Table 4.6 gives an overview of the DBpedia 3.2 knowledge base and ontology which we use in the experimental settings.

The DBpedia ontology classes mainly cover named entities. However, our aim is to annotate all words in text, not only named entities. We therefore use, aside from the ontology, one of the three classification schematas for things provided by the DBpedia project. The three schematas are *Wikipedia categories*, *WordNet synset links* and the *YAGO classification* which is derived from Wikipedia categories and WordNet. We choose Wikipedia categories, which were previously used for measuring concept relatedness (Strube & Ponzetto, 2006), disambiguating named entities (Bunescu & Pasca, 2006) or building a Wikipedia-based taxonomy (Ponzetto & Strube, 2007). DBpedia resources are assigned one or more categories, with a resource having, on average, 3.62 categories. These categories form a hierarchy and are organized as a direct acyclic graph.

We represent the DBpedia knowledge base as a graph $G_D = (V_D, E_D)$, where V_D is the set of all elements including resources, ontology classes and Wikipedia categories and E_D denotes all the relations between these elements. In this graph we identify two subgraphs: a *class subgraph* and a *category subgraph*. The class subgraph $G_{Dc} = (V_{Dc}, E_{Dc})$, $G_{Dc} \subset G_D$ includes all the DBpedia ontology classes

Table 4.5: The DBpedia concepts represented by the resources `http://dbpedia.org/resource/Bus` and `http://dbpedia.org/resource/Bus_(computing)`.

Example concept	*Bus* (`http://dbpedia.org/resource/Bus`) [...] a road vehicle designed to carry many passengers [...]
rdfs:label	Bus
rdfs:comment (short abstract)	A bus is a road vehicle designed to carry many passengers. [...]
dbpedia-owl:abstract (long abstract)	A bus is a road vehicle designed to carry many passengers. Buses can have a capacity as high as 300 passengers.[...]
dcterms:subject	category:Busses category:Cab_over_vehicles category:French_inventions
rdf:type	Thing
Example concept	*Bus_(computing)* (`http://dbpedia.org/resource/Bus_(computing)`) [...] a communication system that transfers data between components inside a computer, or between computers [...]
rdfs:label	Bus_(computing)
rdfs:comment (short abstract)	In computer architecture, a bus is a communication system that transfers data between components inside a computer, or between computers. [...]
dbpedia-owl:abstract (long abstract)	In computer architecture, a bus is a communication system that transfers data between components inside a computer, or between computers. This expression covers all related hardware components and software, including communication protocols. [...]
dcterms:subject	category:Computer_buses category:Digital_electronics category:Motherboard
rdf:type	yago:ComputerBuses

(V_{Dc}) and the relations between these classes as well as between classes and instances (E_{Dc}). Similarly, the category subgraph $G_{Dk} = (V_{Dk}, E_{Dk})$, $G_{Dk} \subset G_D$ includes all the Wikipedia categories (V_{Dk}) and the relations between these categories as well as between categories and resources (E_{Dk}).

In this work we refer to all DBpedia knowledge base resources as concepts which we use to annotate words in context. In order to determine the relatedness between concepts the algorithms integrated in the annotation framework use either the entire DBpedia knowledge base graph G_D or one of the two subgraphs G_{Dc} or G_{Dk}.

Figure 6.3 shows the distribution of node degrees in DBpedia: Figure 4.5a depicts the degree distribution in the entire knowledge base while Figure 4.5b and Figure 4.5c present the degree distribution for the DBpedia ontology and the Wikipedia category schemata respectively. We note the following:

- In the *DBpedia knowledge base*, more than half of the nodes have degree 4 or less. We do not take into account nodes that have a degree of zero.
- In the *DBpedia ontology*, more than half of the nodes have degree above 750. This is due to the high number of instances of a class, as on average a class

has around 5000 instances.

- In the *Wikipedia category schemata*, more than half of the nodes have degree 7 or less. As links representing schemata relations we include relations between two categories or between a category and a resource; we do not take into account nodes that have a degree of zero.

Table 4.6: An overview of the DBpedia 3.2 ontology and knowledge base.

DBpedia Project Version 3.2	
Ontology classes	295
Relations between classes	257
Ontology instances	1,477,796
Knowledge base resources	3,129,565
Categories	590,986
Relations between categories	1,117,715
Resources with categories	2,951,606

4.3.2 Illustrative Example

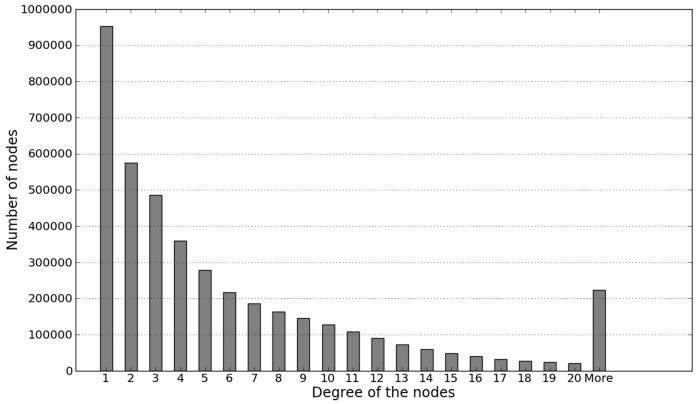
Figure 4.6 shows three possible annotations for the word "bus". These annotations are represented by two DBpedia knowledge base resources. For each of these resources we show the DBpedia ontology class (marked by the *rdf:type* predicate) and two of its Wikipedia categories (marked by the *dcterms:subject* predicate).

4.4 Summary

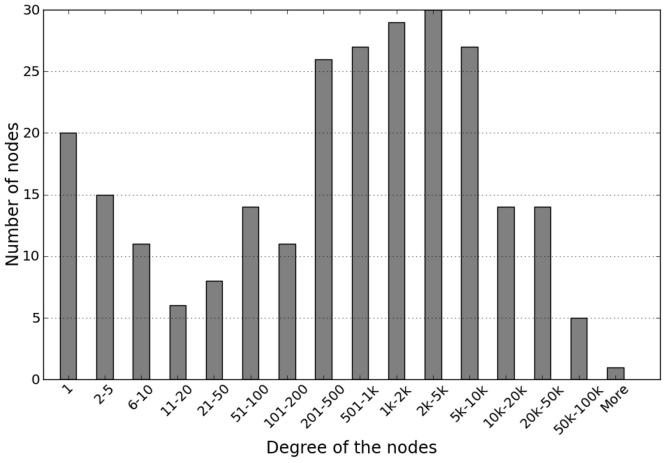
Table 4.7 systematizes the characteristics of the WordNet, OpenCyc and DBpedia ontologies and knowledge bases from the concept relatedness and text annotation perspectives. As far as concepts and relations are concerned, we consider the following when determining the relatedness between concepts and performing text annotation.

Concepts. In WordNet the concepts are represented via instances of the *Synset* and *WordSenses* classes, where a synset contains one or more word senses (see Section 4.1). There are around 117,000 synsets and 206,000 word senses in WordNet (see Table 4.2). In OpenCyc concepts are represented via classes, instances and object properties (e.g. the word "friend"); we consider around 176,000 such concepts (see Table 4.4). For DBpedia we identify three cases, depending on whether we take into account the entire DBpedia knowledge base, the DBpedia ontology or the Wikipedia category schemata. The concepts are represented by all of DBpedia's resources, ontology classes and Wikipedia categories or a subset of the aforementioned elements (see Section 4.3). There are around 3.1 million resources, 290 ontology classes and around 590,000 Wikipedia categories (see Table 4.6).

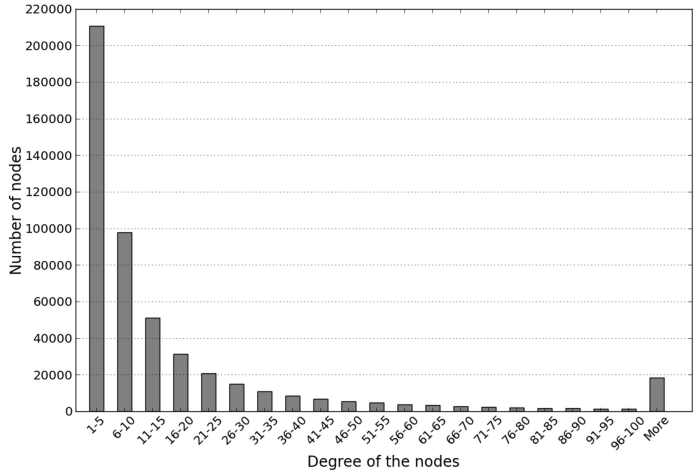
Relations. For WordNet we take into account all available relations, around 377,000 relations between synsets and between word senses (see Section 4.1.1 and Table 4.2). In the case of OpenCyc, we make use of the most common relations defined in the ontology: *rdf:type*, *rdfs:subClassOf* and *broaderTerm*. We consider both *rdf:type* relations between an instance and a class and *rdf:type* relations between two



(a) DBpedia Knowledge Base



(b) DBpedia Ontology



(c) Wikipedia Category schemata

Figure 4.5: The distribution of node degrees in (a) the DBpedia knowledge base, (b) the DBpedia ontology and (c) the Wikipedia category schemata.

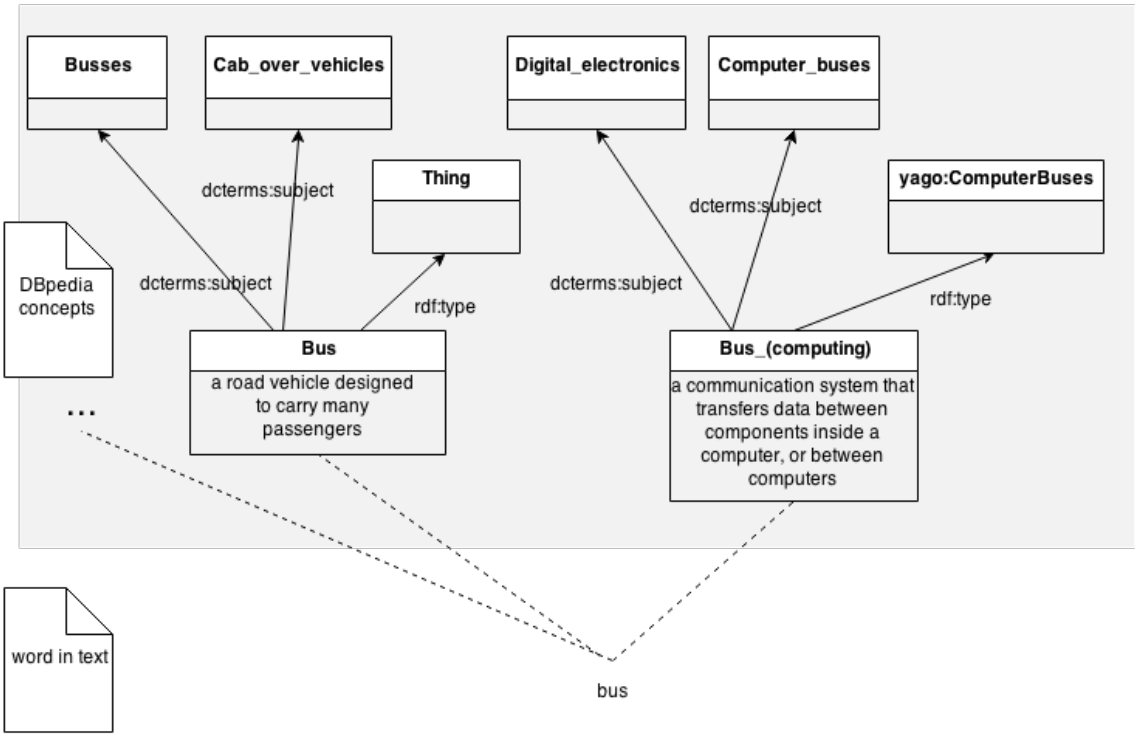


Figure 4.6: Two possible annotations for the word "bus" in the DBpedia knowledge base: *Bus* or *Bus_ (computing)*. The *rdf:type* property states that the concept is an instance of a class while the *dcterms:subject* property relates the concept to its category.

classes; the latter type of relations is used for meta-modeling, providing additional structure to the ontology. *BroaderTerm* relations are defined between concepts that are not strictly taxonomic. However, our approach is not restricted to only these types of relations. All in all we take into account around 423,000 relations (see Table 4.4). For DBpedia we consider all relations defined between concepts, where the concepts are represented as discussed above. The entire DBpedia knowledge base graph has around 8.4 million relations; in the case of the DBpedia ontology we are considering around 250 relations between classes while for the Wikipedia category graph we represent around 1.1 million relations between categories (see Table 4.6).

Table 4.7: Characteristics of WordNet, OpenCyc and DBpedia which affect concept relatedness and text annotation.

	WordNet	**OpenCyc**	**DBpedia**
Purpose of the ontology or knowledge base	Lexical database containing concepts which represent words and collocations	General purpose ontology	Structured representation of Wikipedia's encyclopedic knowledge
How are concepts specified	Via instances of the *Synset* and *WordSenses* classes (a synset contains one or more word senses)	Via classes, instances, object properties (see, for e.g. word "friend")	Many resources (instances) organized in a shallow cross-domain ontology
Number of abstract concepts	Concepts mainly correspond to lexical terms and collocations, the number of abstract concepts being low	Several abstract concepts for grouping information (e.g. the concept *Collection*)	Several abstract concepts for grouping information (e.g. the concepts *Activity* or *Agent*)
Concept definitions	In the form of *glosses*	Only for 37% of the classes, instances and object properties	In the form of *short* and/or *long abstracts*
Example sentences containing concepts	Yes	No	No

Chapter 5

Automatic Text Annotation Framework

This chapter is based on the work presented in (Rusu & Mladenić, 2014) and describes an *Automatic Text Annotation Framework* which uses information represented in an ontology or knowledge base as a source of background knowledge.

The proposed modular framework annotates text with concepts defined in a knowledge base and relies on a graph-based representation of the knowledge base. The framework comprises two main modules (see Figure 5.1):

- a *concept relatedness module* which, given two concepts defined in the ontology or knowledge base, outputs the relatedness between these concepts;
- a *text annotation module* which, given a text fragment and an ontology or knowledge base as input, annotates each word or collocation with concepts from the ontology or knowledge base, relying on the relatedness between concepts.

The framework modularity enables the integration of various relatedness measures for ranking candidate concepts, which take into account different characteristics of the knowledge base.

In order to annotate a text fragment, we first pre-process the text using standard techniques: identifying words and collocations, lemmatization and part-of-speech tagging (see Section 5.2.1). Secondly, we determine, for each word or collocation, a set of candidate concepts; this step depends on the knowledge base used as concept inventory. For example, in the case of WordNet this implies identifying candidate concepts based on string matching between the word or collocation lemmas and the concept labels, given the word's part-of-speech. In the case of DBpedia, redirects and disambiguation links help identifying candidate concepts. Section 5.2.2 describes the general approach to candidate concept identification while its application to specific knowledge bases is presented in Chapter 4. The text annotation algorithm selects from the set of candidate concepts the concept that most appropriately matches the context. This can be seen as a concept ranking problem, where the candidate concepts are ranked based on how related they are to their context (see Section 5.2.3). Finally, the candidate concepts obtaining the best score are chosen as annotations for the words or collocations in the text fragment (see Section 5.2.4).

The relatedness between concepts is determined using the relatedness measures proposed in Chapter 3, which rely either on concept definitions, the knowledge base

structure or a combination of the two. Additionally, the generality of our approach allows us to integrate other relatedness measures described in the literature.

In what follows we present in more detail the two main modules of the text annotation framework: the *relatedness module* and the *text annotation* module.

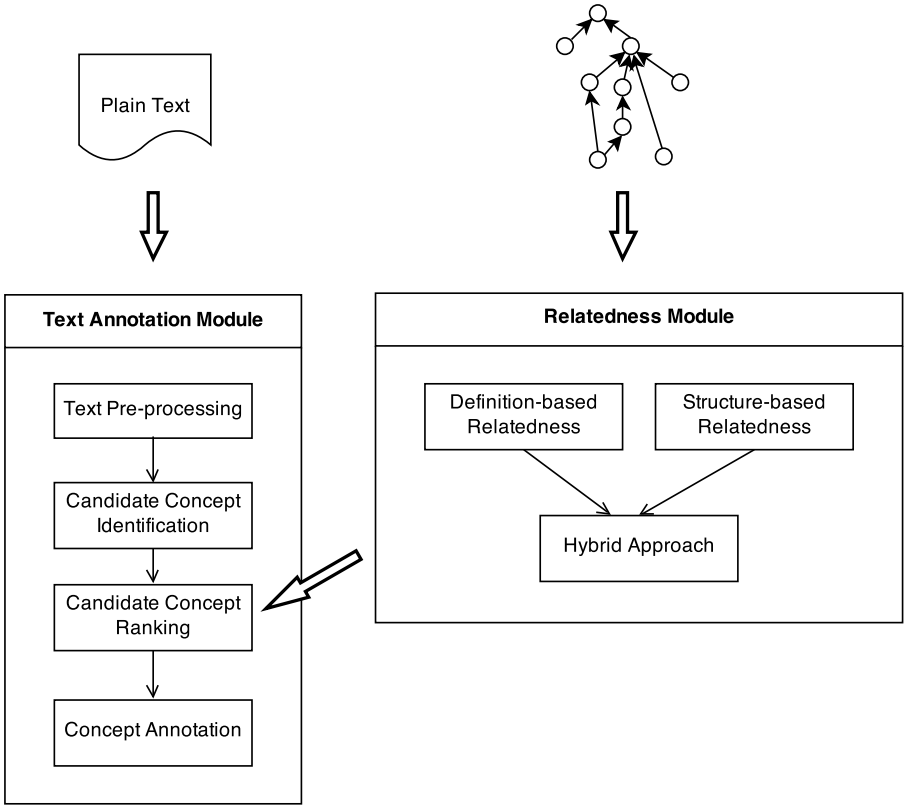


Figure 5.1: The proposed text annotation framework. The input is a plain text fragment to annotate and the knowledge base represented as a graph. The relatedness measure based either on the knowledge base structure, concept definitions or a combination of the two is used for ranking candidate concepts.

5.1 Relatedness Module

The *Relatedness Module* implements the three types of relatedness measures presented in Chapter 3. These are the definition-based relatedness measure, a relatedness measure which relies on the knowledge base structure and the hybrid approach which combines the two aforementioned measures.

Given a graph representation $G = (V, E)$ of a knowledge base, with V as the set of all knowledge base concepts and E as the set of relations defined between these concepts, a pair of concepts v_1 and v_2 and the type of relatedness measure, the module outputs the relatedness between this pair of concepts $Relatedness(v_1, v_2)$. To reduce computational complexity, the relatedness module can pre-compute pairwise relatedness measures for the knowledge base concepts under consideration.

Aside from the measures defined in Chapter 3, one can integrate in the proposed Relatedness Module any other relatedness measure which is defined for a pair of knowledge base concepts.

5.2 Text Annotation Module

The *Text Annotation Module* integrates four main components: a *Text Pre-processing* component which provides an internal representation of the unstructured text fragment received as input; a *Candidate Concept Identification* component which, given a word or collocation as input, identifies a set of candidate concepts defined in the knowledge base; a *Candidate Concept Ranking* component which ranks candidate concepts based on a relatedness score and a *Concept Annotation* component which assigns to each word or collocation the best ranked concept belonging to the set of corresponding candidate concepts.

5.2.1 Text Pre-processing

The goal of the *Text Pre-processing* component is to generate an internal representation of the input unstructured text fragment. More formally, given an input text T, the pre-processing component generates an output sequence $W = (w_1, w_2, ..., w_N)$, where w_i represents a word or collocation identified in the input text and N denotes the total number of words or collocations [1]. For each word w_i we identify the sentence that it belongs to, its lemma, part-of-speech and named entity type. A *stop words* list is used to filter out words that are not useful for text annotation; an example would be function words such as *the* or *a*. Stop words are removed only after collocations are identified.

In our experiments we use the pre-processing tools implemented in NLTK (Bird, Klein, & Loper, 2009) and Stanford CoreNLP (Toutanova, Klein, Manning, & Singer, 2003; Finkel, Grenager, & Manning, 2005).

Identifying words and collocations. Sentence boundaries are identified via a *sentence splitter* and a *tokenizer*[2] is used to obtain a set of tokens for each sentence. In order to detect collocations we use the lemmatized tokens to build candidate n-grams which we then match with a list of frequent collocations (in our experiments we consider bigrams and trigrams). For obtaining frequent collocations we use NLTK's collocation module [3]. If we identify a collocation which does not have a corresponding concept in the ontology, then this collocation will not be annotated even if words from the collocation appear in the ontology.

Lemmatization. In English words have several inflected forms; the word lemma is the base form of the word. We use WordNet's *morphy* function to lemmatize words.

Part-of-speech Tagging. Part-of-speech taggers part of Stanford CoreNLP or NLTK [4] are used to assign to each word its corresponding part-of-speech.

Named entity recognition. The most common named entities such as people, locations and organizations are identified by Stanford CoreNLP's named entity recognition tool.

5.2.2 Candidate Concept Identification

For each word to annotate $w_i \in W$ the *Candidate Concept Identification* component determines a set of *candidate concepts* $C_i = \{c_{i,1}, c_{i,2}, ..., c_{i,m_i}\}$ defined in the

[1] In the remainder of this chapter the term *word* will refer to either a single word or a collocation.
[2] nltk.tokenize `http://www.nltk.org/api/nltk.tokenize.html`
[3] nltk.collocations `http://www.nltk.org/_modules/nltk/collocations.html`
[4] nltk.tag `http://www.nltk.org/api/nltk.tag.html`

knowledge base which can be valid annotations for w_i. In the most general case, this step is based on string matching between pairs of surface forms: the lemma of w_i and natural language identifiers (NLI) of concepts from the knowledge base. We also lemmatize the concept NLI which has to match exactly with w_i's lemma. Figure 5.2 shows an example WordNet and DBpedia candidate concepts for the word *ministers*. WordNet concepts are grouped based on their part-of-speech; knowing the word part-of-speech helps narrowing down the number of candidate concepts. In our example, knowing that the word *ministers* is a *noun* excludes the two verbs from the set of candidate concepts.

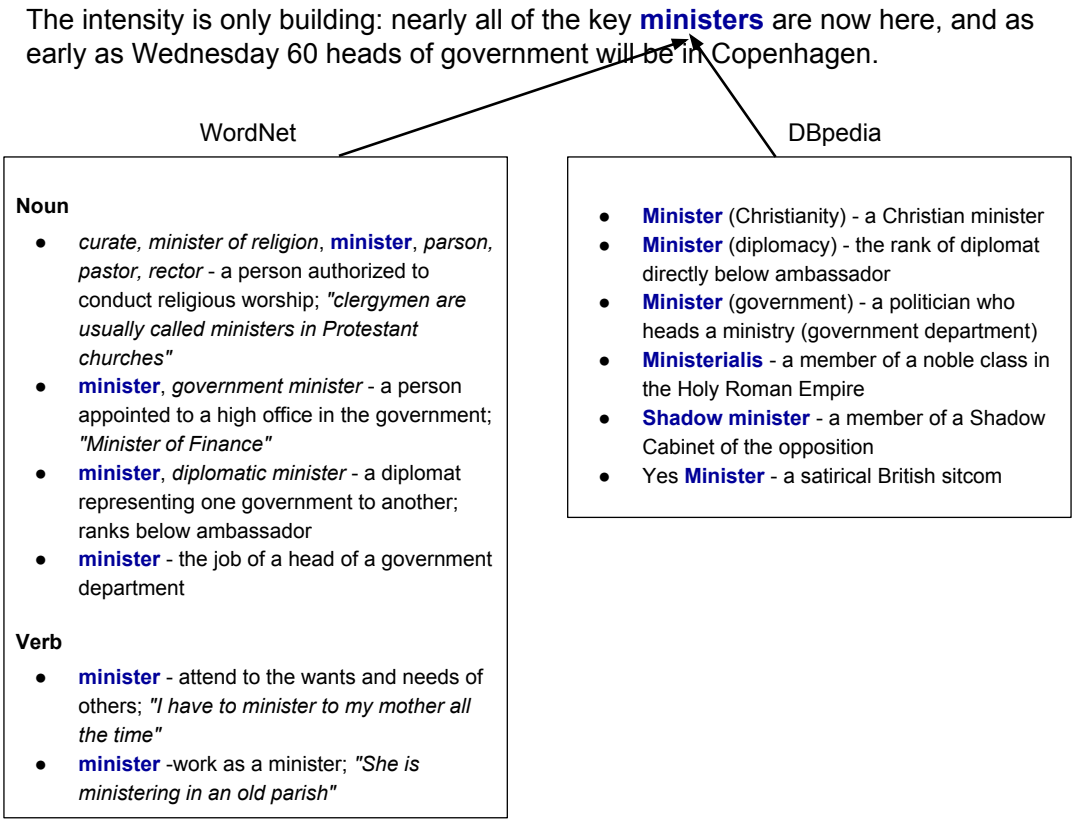


Figure 5.2: Example WordNet and DBpedia candidate concepts for the word *ministers*, obtained by matching the word lemma *minister* with the natural language identifiers (NLIs) of WordNet and DBpedia concepts. The NLIs for each concept are marked in *italic* and the matching NLI is highlighted.

5.2.2.1 WordNet

In order to determine candidate concepts for a given word or collocation we search WordNet synsets and retrieve a subset of synsets which constitute the candidate concepts. This subset of synsets is identified based on string matching between the word lemma and the synset word senses, given the word part-of-speech. For example, we can identify two candidate concepts for the noun "bus" as the word lemma matches one of the word senses of the corresponding synsets *autobus* and *busbar* (see Table 4.1).

5.2.2.2 OpenCyc

Candidate concept identification in the case of OpenCyc consists of retrieving a subset of concepts which best match the word or collocation to annotate. We start by identifying a set of strings which include the word or collocation surface forms as well as the corresponding lemma. Next, we obtain candidate concepts by matching this set of strings to concept natural language identifiers (via *rdfs:label* or *prettyString*). For example, we can identify two candidate concept for the word "bus", namely *Bus-RoadVehicle* and *ComputerBus* as both the *rdfs:label* and *prettyString* match the word surface form (see Table 4.3).

5.2.2.3 DBpedia

Following Bunescu and Pasca (2006) we determine candidate concepts from the entire DBpedia knowledge base by taking into account *redirects* and *disambiguation links*. Redirects link resources with alternative names. For example the triplet:

```
<http://dbpedia.org/resource/AxiomOfChoice>
<http://dbpedia.org/ontology/wikiPageRedirects>
<http://dbpedia.org/resource/Axiom_of_choice>
```

links *AxiomOfChoice* to its alternative name *Axiom_ of_ choice*. Disambiguation links are used to group ambiguous names. For example:

```
<http://dbpedia.org/resource/Austin_(disambiguation)>
```

links to 27 disambiguated resources like:

`<http://dbpedia.org/resource/Austin,_Texas>` or
`<http://dbpedia.org/resource/University_of_Texas_at_Austin>`.

Similar to OpenCyc, we first identify a set of strings which include the surface form and lemma or the word or collocation to annotate. Next, we obtain candidate concepts by matching this set of strings to concept natural language identifiers (NLIs) via the *rdfs:label* and redirect links. This gives us the initial set of candidate concepts, which we augment with all the corresponding disambiguation links. For example, the initial set of candidate concepts for the word "bus" contains the DBpedia concept *{Bus}* obtained by matching the word with the concept NLI. This set is augmented with the corresponding disambiguation links *{Bus, Bus_ (computing), etc.}* and redirect links *{Autobus, Charter_ Bus, etc.}*. This yields a final set of candidate concepts formed of *{Bus, Bus_ (computing), Autobus, Charter_ Bus, etc.}*.

5.2.3 Candidate Concept Ranking

The *Candidate Concept Ranking* component ranks the candidate concepts of each word w_i based on the relatedness measure between these concepts and the *local context* of w_i. The *local context* of w_i is represented by its neighboring words within a variable-sized window. A typical window consists of $2k$ words, k of them before and k after w_i. In some cases, e.g. at the beginning or end of the text fragment, there might not be k words preceding or following the word to annotate. If the number of

words before (or after) w_i is less than k and there are more words after (or before) w_i, then these words are also included so that the number of words in the local context is as close as possible to $2k$. Additionally, as the same word can occur multiple times in the text sequence, we make sure to exclude from the local context any duplicate occurrence of w_i. Let $L_i = \{i-k, i-k+1, ..., i-1, i+1, ..., i+k-1, i+k\}$ be a set representing the indices of words in the local context of w_i. For each word w_i to annotate we first identify the corresponding set of candidate concepts $C_i = \{c_{i,1}, c_{i,2}, ..., c_{i,m_i}\}$. Similarly, we obtain the set of candidate concepts corresponding to the local context L_i of w_i, denoted by $C_{L_i} = \bigcup_{j \in L_i} C_j$. Next, we determine the pairwise relatedness between all concept candidates in C_i and C_{L_i}, $R(c_p, c_t)$, with $c_p \in C_i$ and $c_t \in C_{L_i}$. For each candidate concept $c_p \in C_i$ we aggregate all corresponding relatedness values; the candidate concepts in C_i are ranked based on the aggregated relatedness score, where the candidate concept with the maximum aggregated relatedness score is defined as:

$$\hat{c_p} = \underset{c_p \in C_i}{\operatorname{argmax}} \operatorname{agg}_{c_t \in C_{L_i}} R(c_p, c_t) \tag{5.1}$$

The agg function in Eq. 5.1 is an aggregate function. We evaluate the performance of our text annotation algorithm using three such aggregate functions: *maximum*, *average* and *median* (see Chapter 6).

5.2.4 Text Annotation

As a final step, for each word w_i the *Text Annotation Component* assigns a corresponding annotation a_i which is represented by the candidate concept with the maximum aggregated relatedness score $\hat{c_p}$. The output of this component is therefore a sequence of annotations $A = (a_1, a_2, ..., a_N)$ which correspond to the input text sequence $W = (w_1, w_2, ..., w_N)$.

Algorithm 5.1 summarizes the text annotation algorithm which maps an input text sequence W to a sequence of annotation concepts A defined in a knowledge base.

Figure 5.3 shows the steps performed by the text annotation algorithm for assigning concepts to five words. Assume we want to annotate w_1 for which we identify three candidate concepts: $C_1 = \{c_{1,1}, c_{1,2}, c_{1,3}\}$. The local context of w_1 includes words w_2 through w_5, therefore $L_1 = \{2, 3, 4, 5\}$, while the set of candidate concepts for the local context is $C_{L_1} = \bigcup_{j \in L_1} C_j$, a total of ten concepts (2 for w_2, 3 for w_3, 1 for w_4 and 4 for w_5). We determine the pairwise relatedness for each pair of concepts (c_p, c_t) with $c_p \in C_1$ and $c_t \in C_{L_1}$ and aggregate the relatedness values for each $c_p \in C_1$. The concept with the highest aggregated relatedness score, in this example c_2, is chosen to annotate w_1. In *step2* we focus on w_2 which has two candidate concepts and a set of candidate concepts corresponding to the local context composed of nine concepts. Note that the local context size is shrinking as more concepts are annotated. Moreover, once a concept was annotated according to the evidence provided by its local context this annotation does not get updated. By fixing the local window size we assume that only concepts belonging to the local context are relevant for selecting the annotation concept. If more concepts turn out to be relevant, they can be taken into account by increasing the window size. The annotation algorithm continues to assign concepts for words w_3, w_4 and w_5 in steps 3 and 4 respectively. If a word has only one candidate concept, like the case

Algorithm 5.1: The text annotation algorithm maps an input text sequence $W = (w_1, w_2, ..., w_N)$, where N denotes the number of words or collocations to a sequence of annotation concepts $A = (a_1, a_2, ..., a_N)$ defined in a knowledge base.

Data: $G(V, E)$ knowledge base graph representation
$W = (w_1, w_2, ..., w_N)$ text sequence
$2k$ local context size
Result: $A = (a_1, a_2, ..., a_N)$ sequence of annotations

for $i = 1$ *to* N **do**
  `/* local context indices */`
  $L_i = \{i-k, i-k+1, ..., i-1, i+1, ..., i+k-1, i+k\}$
  `/* candidate concepts for the word to annotate */`
  $C_i = \{c_{i,1}, c_{i,2}, ..., c_{i,m_i}\}$
  `/* candidate concepts in the local context */`
  $C_{L_i} = \bigcup_{j \in L_i} C_j$
  `/* determine the relatedness between the candidate concepts and the local context */`
  for $c_p \in C_i$ **do**
    for $c_t \in C_{L_i}$ **do**
      $R(c_p, c_t) = Relatedness(c_p, c_t)$
    end
  end
  $\hat{c_p} = \text{argmax}_{c_p \in C_i} \text{agg}_{c_t \in C_{L_i}} R(c_p, c_t)$
  $a_i = \hat{c_p}$
end

of w_4, then we assign this concept to the word and continue with the next word to annotate.

The intuition behind our approach is that the local context of each word contains evidence that helps to annotate that word. As we show in the evaluation section, the size of the local context depends on the text to annotate and the ontology used as concept inventory. A small window size might not contain enough relevant concepts to provide a good annotation, whereas a window size that is too wide might bring about too much noise and therefore wrong annotations. For example, consider the sentence in Figure 5.2. The word **ministers** can be annotated with six DBpedia concepts: *Minister (Christianity), Minister (diplomacy), Minister (government), Ministerialis, Shadow minister (Shadow Cabinet)* and *Minister (sitcom)*. In order to choose the correct annotation for **ministers** which is **Minister (government)** the most indicative collocation is **heads of government**.

5.3 Summary

This chapter defined the automatic text annotation framework with its two main modules, the relatedness module described in Section 5.1 and the text annotation module described in Section 5.2.

In the next chapter (Chapter 6) we present in more detail the evaluation settings for the relatedness measures and the text annotation framework.

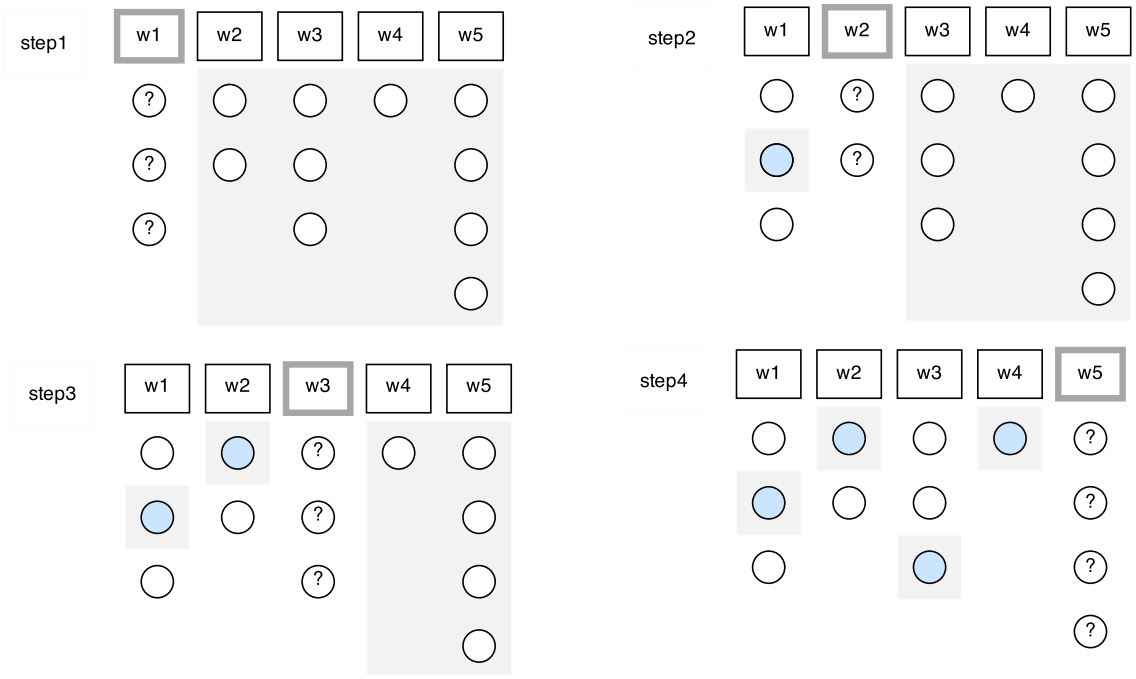


Figure 5.3: Steps performed by the text annotation algorithm for assigning concepts to five words. At each step we mark with ? the candidate concepts for the word to annotate and with a gray-shaded area the local context for that word. As words are assigned concepts the size of the local context shrinks. Note that w_4 has only one candidate concept, in which case no concept ranking is required.

Chapter 6

Evaluation

This chapter describes the experimental settings and results for the proposed relatedness measures (see Section 6.1) and for the text annotation framework (see Section 6.2). As background knowledge we use WordNet, OpenCyc and DBpedia for measuring the relatedness between concepts, and WordNet and DBpedia for text annotation, respectively.

6.1 Relatedness Measures

In this section we evaluate the performance of the proposed relatedness measures and present experiments on three different knowledge bases: WordNet, OpenCyc and DBpedia. We start by describing the datasets used in the evaluation settings, followed by an explanation of the evaluation metrics and the results that we obtained for each knowledge base.

We compare our proposed approaches to measuring relatedness (see Chapter 3) to various algorithms from the literature as described in the related work section (see Chapter 2). We have re-implemented some of those algorithms, in order to apply them to the knowledge bases used in the evaluation settings. Table 6.1 provides a short summary of the re-implemented approaches, and the knowledge bases they have been applied to.

6.1.1 Evaluation Dataset Description

For all three knowledge bases, namely WordNet, OpenCyc and DBpedia we make use of three standard evaluation datasets that have been previously applied for comparing different similarity and relatedness measures. Additionally, we perform an evaluation on a subset of OpenCyc concepts, and propose a clustering approach for validating the results.

6.1.1.1 Standard Datasets

For assessing the performance of our approach, we consider three standard datasets that have been previously used for evaluating similarity and relatedness measures based on the WordNet lexical database (Agirre, Alfonseca, Hall, Kravalova, & Pas, 2009; Schwartz & Gomez, 2011).

The first dataset, RG, proposed by Rubenstein and Goodenough (1965) consists of 65 word pairs which were assigned scores between 0.0 and 4.0 by 51 human asses-

Table 6.1: A short summary of the re-implemented approaches used in the evaluation settings.

Approach Name	**Knowledge Base**	**Description**
Wu and Palmer	WordNet, OpenCyc	The method is based on determining the least common subsumer of the two concepts.
Leacock and Chodorow	WordNet, OpenCyc	This method scales the distance between two concepts with the depth of the taxonomy.
Adapted Google Distance	DBpedia	This method is similar to the one proposed by Milne and Witten (2008a), but differs in that we take into account all relations between two concepts as opposed to considering only Wikipedia page links.
Shortest Path Unit Weight	WordNet, OpenCyc, DBpedia	This method determines the distance between two concepts by applying a shortest path algorithm on a unit-weighted graph. $RW(v_i, v_j) = 1$, where RW represents the relation weight.
Moore et al.	WordNet, OpenCyc, DBpedia	This method determines the distance between two concepts by applying a shortest path algorithm on a weighted graph. The edge weights are obtained by summing up the logarithm of the node degrees v_i and v_j. $RW(v_i, v_j) = \log(Degree(v_i)) + \log(Degree(v_j))$, where RW represents the relation weight.

sors. Their judgment was only based on the similarity between the word pairs, all other relationships being disregarded. The second dataset, MC (Millers & Charles, 1991), consists of a 28-word pair subset of the RG dataset, and was used for validating the results obtained in Rubenstein and Goodenough (1965). The third dataset, WordSim353 (Finkelstein et al., 2010) contains 353 word pairs, each annotated by 13 to 15 human judgments. Using this dataset, Agirre et al. (2009) annotated pairs of words with different relationships: identical, synonymy, antonymy, hyponymy, and unrelated. The studies described in Rubenstein and Goodenough (1965), Millers and Charles (1991), Resnik (1995) report high inter-annotator agreements between the human judgment for the RG and MC datasets.

In Schwartz and Gomez (2011), the authors provide WordNet 3.0 concepts for the aforementioned word pairs, and analyze similarity and relatedness measures applied to the word and concept pairs, respectively. In cases where there is no appropriate concept, the word pair is discarded. For the WordSim353 dataset, Schwartz and Gomez did not take into account the pairs marked as unrelated. We choose to evaluate our measures on this dataset, and look at concept pairs rather than word pairs. By doing so, we avoid the ambiguity arising from comparing the similarity and relatedness measures with human judgments on word pairs.

For our OpenCyc experiments we map the WordNet 3.0 concepts provided in (Schwartz & Gomez, 2011) to OpenCyc concepts, and discard pairs where at least one concept is not present in OpenCyc. Some WordNet concepts are mapped to OpenCyc object properties. The mapping was performed by two annotators, with a Cohen's kappa coefficient of inter-annotator agreement of 0.750 (Cohen, 1960). We

obtain 20 concept pairs for the Millers and Charles dataset, 51 concept pairs for the Rubenstein and Goodenough dataset and 71 concept pairs for the WordSim dataset.

A similar mapping is performed for the DBpedia experiments, where for the aforementioned WordNet 3.0 concepts we identify matching DBpedia concepts. In the case of this mapping we report a Cohen's kappa coefficient of inter-annotator agreement of 0.82. We obtain 24 concept pairs for the Millers and Charles dataset, 59 concept pairs for the Rubenstein and Goodenough dataset and 85 concept pairs for the WordSim dataset.

6.1.1.2 Subset of OpenCyc Concepts

In addition to evaluating the performance of our algorithms on the previously-mentioned standard datasets, we propose a clustering approach for validating the results on a subset of OpenCyc concepts. The aim is to show that our proposed algorithm relying on weighted concept paths can also be used for clustering concepts based on the similarity between them.

Our synthetic data consists of 108 randomly chosen words belonging to four different categories: 24 names of countries, 35 names of fruits, 21 of computer software and 28 of hardware. Each word is mapped to one or more OpenCyc concepts. For example, the word "apple" is mapped to the OpenCyc concepts *Apple* (the fruit) and *AppleInc* (the software company). Countries are mainly represented as instances in OpenCyc, while names of fruits, computer hardware and software are mainly represented as classes.

6.1.2 Evaluation Metrics

We use different evaluation metrics depending on the dataset used as input. In the case of standard datasets which were manually labeled by human assessors we use the Spearman rank correlation as the evaluation metric. For the clustering experiment on a subset of OpenCyc concepts, on the other hand, we use standard internal clustering evaluation techniques for validating the results.

6.1.2.1 Standard Datasets

In the evaluation setting based on standard datasets we report Spearman rank correlations between human judgment and various algorithms for determining concept similarity and relatedness. Spearman's rank correlation is preferred to the Pearson correlation in cases where no linear relationship between two random variables can be expected (Agirre et al., 2009).

The *Spearman rank correlation coefficient* ρ measures the statistical dependence between two variables. An absolute value of $\rho = 1$ indicates full agreement between the human judgment and the relatedness algorithms. The Spearman correlation coefficient is defined as:

$$\rho = \frac{\sum_i (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_i (x_i - \overline{x})^2 \sum_i (y_i - \overline{y})^2}}, \tag{6.1}$$

where x_i and y_i are the ranks corresponding to the scores X_i and Y_i given by the human judgment and the relatedness algorithms, respectively. If the algorithm assigns identical scores to two or more pairs of concepts, their corresponding rank equals the average of their positions.

We test whether a relatedness algorithm yields an output which is correlated with human judgment. Therefore we try to reject the null hypothesis of no correlation, under which the test statistic t:

$$t = r\sqrt{\frac{n-2}{1-r^2}}, \tag{6.2}$$

is approximately *Student's t distributed* with $n-2$ degrees of freedom, r being the sample rank correlation and n being the sample size.

6.1.2.2 Subset of OpenCyc Concepts

In order to validate the results obtained on a subset of OpenCyc concepts, we propose a clustering approach. We make use of standard internal clustering evaluation techniques for validating the results: the *intra-cluster distance*, the *inter-cluster distance* and the *Davies-Bouldin Index* (Davies, 1979). In our case, the intra-cluster distance or scatter is a measure characterizing the concept distance between members of the same cluster, and should be as low as possible. The inter-cluster distance or the separation between clusters characterizes the concept distance between members of different clusters, and should be as large as possible. The Davies-Bouldin Index (DBI) is defined as the ratio of the scatter within a cluster to the separation between clusters; good clustering algorithms have a low DBI value.

The DBI relies on clusters of vectors; for each cluster a centroid can be determined. As in this case we are dealing with pairwise distances between concepts, we define a *modified DBI* having the cluster scatter S_i and the separation between clusters $M_{i,j}$ depending on these distances as follows:

$$S_i = \sqrt[q]{\frac{2}{N_i(N_i-1)}\sum_{k=1}^{N_i}\sum_{p=1}^{k-1} DS(c_k, c_p)^q}, \tag{6.3}$$

and

$$M_{i,j} = \sqrt[q]{\frac{1}{N_i N_j}\sum_{k=1}^{N_i}\sum_{p=1}^{N_j} DS(c_k, c_p)^q}, \tag{6.4}$$

where N_i is the number of concepts in cluster i and $DS(c_k, c_p)$ is the distance between the c_k and c_p concepts. The scatter S_i is determined based on the distance between concepts c_k and c_p belonging to the same cluster i. The separation between clusters $M_{i,j}$ is based on the distance between concepts c_k and c_p belonging to different clusters i and j. The value of q is usually 2, corresponding to the *Euclidean distance*.

6.1.3 WordNet

Table 6.2 reports Spearman rank correlations between human judgment and different similarity and relatedness measures. The concept definition-based relatedness measure is referred to as *WordNet Definition*, the structure-based relatedness measures are referred to as *WeightedConceptPath Log* and *Sqrt*, respectively, and the *Hybrid* measure denotes the combination of the aforementioned measures. The relatedness measures have been adapted to obtain similarity measures by restricting the relations to include only taxonomic ones.

Despite the small sample sizes, $p < 10^{-5}$ for all systems evaluated in Table 6.2 allow to decisively reject the null hypothesis of no correlation between the systems and the human judgment of similarity.

For all three standard datasets, the best results are obtained by combining the concept definitions and the knowledge base structure in the hybrid relatedness measure (*Hybrid Log/Sqrt*). Moreover, results improve if we take into account all types of relations defined in the knowledge base as opposed to using only taxonomic relations (marked in Table 6.2 with the word *similarity*).

Table 6.2: Spearman rank correlations between several systems and the human judgments obtained on three standard datasets (MC, RG and WordSim). The measures marked in *italic* were proposed in this thesis. The measures marked with *(similarity)* take into account only taxonomic relations, while the non-marked versions take into account all WordNet 3.0 relationships. The best results obtained by our proposed systems and the re-implemented systems on the one hand, and the results reported in related work on the other hand, are rendered in bold.

Measures used in the evaluation	**MC-WordNet Miller and Charles**	**RG-WordNet Rubenstein and Good-enough**	**WordSim-WordNet Finkelstein et al.**
WeightedConceptPath Log	0.835	0.857	0.667
WeightedConceptPath Log (similarity)	0.785	0.811	0.592
WeightedConceptPath Sqrt	0.833	0.827	0.687
WeightedConceptPath Sqrt (similarity)	0.804	0.801	0.598
WordNet Definition	0.865	0.811	0.689
WordNet Definition (similarity)	0.858	0.820	0.694
Hybrid Log	0.876	**0.862**	0.700
Hybrid Log (similarity)	0.858	0.841	0.705
Hybrid Sqrt	**0.880**	0.856	**0.715**
Hybrid Sqrt (similarity)	0.858	0.843	0.706
Moore et al.	0.808	0.833	0.650
Moore at al. (similarity)	0.792	0.811	0.590
Shortest Path Unit Weight	0.803	0.811	0.601
Shortest Path Unit Weight (similarity)	0.775	0.816	0.570
Spearman rank correlations as reported by Schwartz and Gomez (2011)			
Wu Palmer	0.76	0.79	0.57
Leacock Chodorow	0.75	0.80	0.58
Schwartz Gomez	0.81	0.77	0.54
Resnik	0.76	0.76	0.59
Jiang Conrath	0.85	0.80	0.51
Lin	0.80	0.78	0.58
Hirst St Onge	0.72	0.76	0.53
Yang Powers	0.76	0.78	**0.63**
Banerjee Pedersen	0.76	0.69	0.46
Partwardhan Pedersen	**0.88**	**0.81**	0.55

We make two remarks regarding the concept definition-based relatedness and the hybrid relatedness measures.

Concept definition-based relatedness. For determining the relatedness between two concepts using their concept definitions we associate a definition weight α with each term vector corresponding to the concept (see Eq. 3.1). Recall that each concept has assigned multiple term vectors: for the concept definition and the definitions of connected concepts. In order to assess the influence of the definition weight α on the Spearman rank correlation results we conduct the following experiment. Assume we want to determine the relatedness between two concepts c_1 and c_2. We assign a weight $\alpha = 1$ to the term vector corresponding to the definition of c_1 and c_2, respectively, as this concept definition is the most relevant for determining the degree of relatedness. We test different values of α for term vectors corresponding to definitions of concepts connected with c_1 and c_2. In this experiment we consider two concepts as being connected if there is a direct path between these concepts in the knowledge base graph. We also experimented with paths of higher length, but noted no significant difference in the results, in the case of the WordNet knowledge base. Figure 6.1 shows Spearman rank correlations when varying the definition weight α.

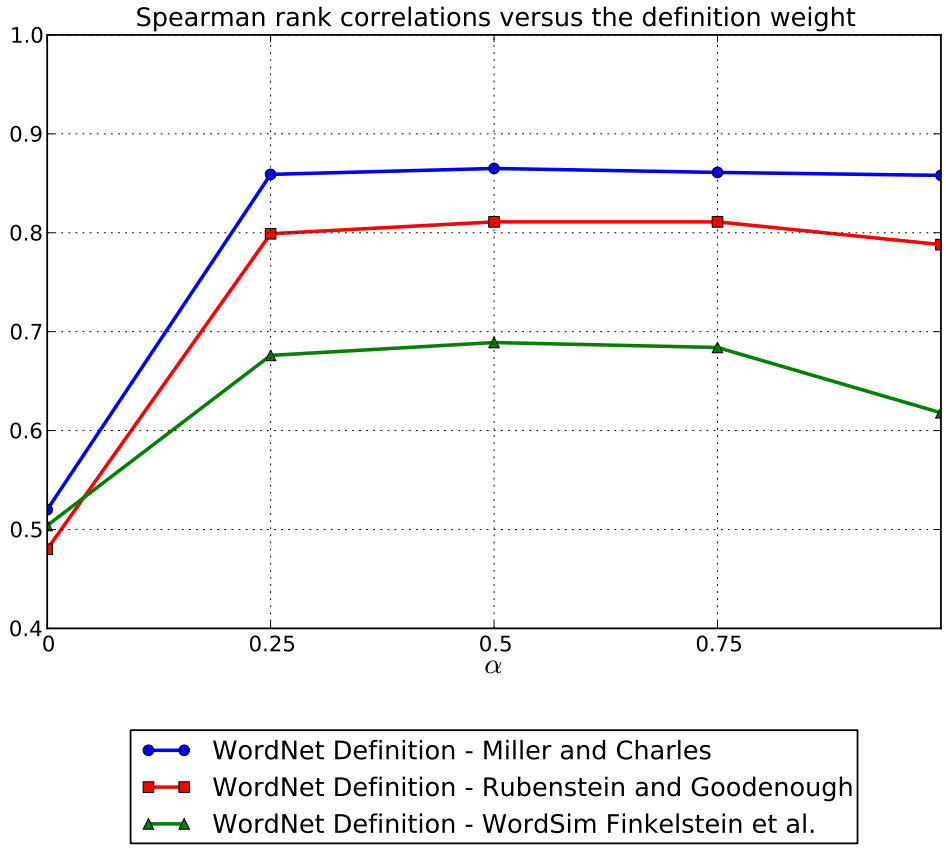


Figure 6.1: Spearman rank correlations for varying definition weight α for WordNet concepts. The results are obtained using the three standard datasets described in the evaluation settings.

Results show that connected concepts are useful for improving the correlation between the *WordNet Definition* system and the human judgment. Moreover, the performance of the concept definition-based system can be further improved by weighting the contribution of connected concepts. This is most notable in the case of the WordSim dataset, which is the largest dataset in terms of concept pairs. In general, a definition weight of $\alpha = 0.5$ assigned to term vectors corresponding to connected concepts yields good results. We therefore report *WordNet Definition* results for this value of α in Table 6.2.

The Hybrid measure of relatedness. This relatedness measure is obtained by weighting the contribution of the concept definition-based relatedness and the structure-based relatedness (see Eq. 3.11). Figure 6.2 depicts different Spearman rank correlations depending on the hybrid weight ζ. As the *WordNet Definition* measure slightly outperforms the structured-based measures *WeightedConceptPath Log/Sqrt*, we obtain best results for $\zeta = 0.6$; these are the results reported in Table 6.2 for the *Hybrid* measure. Additionally, in the case of WordNet, information provided by both the structure and the concept definitions yields the best rank correlations with human judgment.

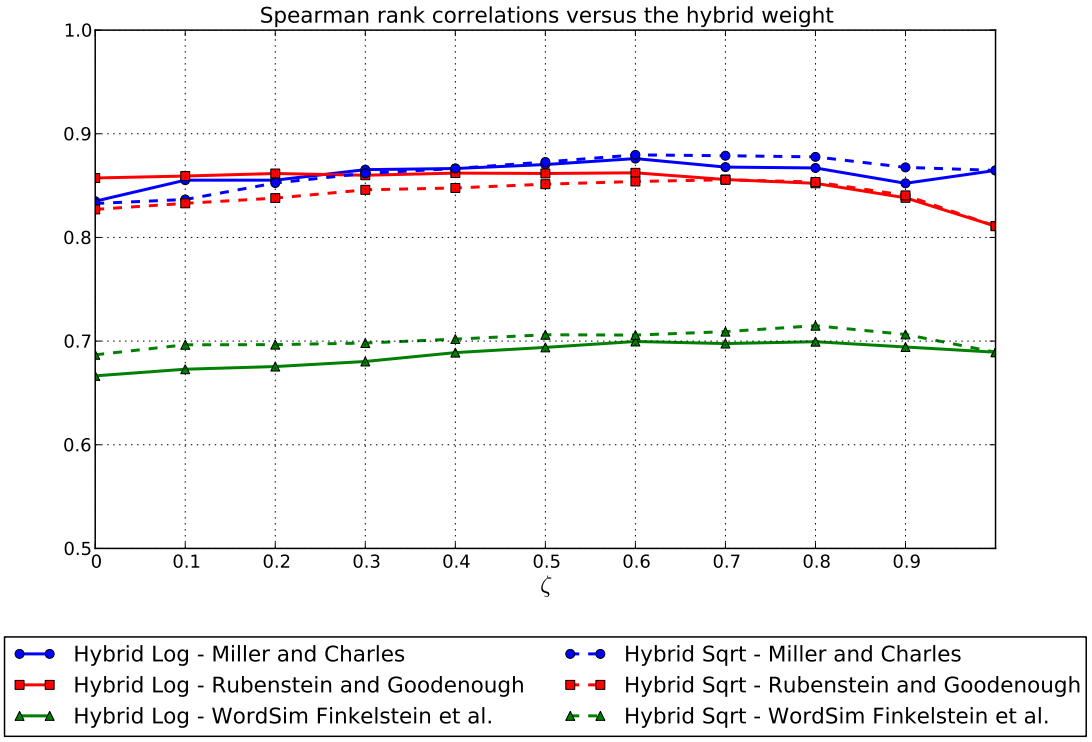


Figure 6.2: Spearman rank correlations for varying hybrid weight ζ for WordNet concepts. The results of the *Hybrid Log* and *Sqrt* measures are obtained using the three standard datasets described in the evaluation settings.

6.1.4 OpenCyc

6.1.4.1 Experiments Using Standard Datasets

Similarly to the WordNet evaluation, in this setting on OpenCyc we report Spearman rank correlations between the human judgment and various algorithms for determining concept similarities. The Spearman rank correlations between the aforementioned systems and human judgment is presented in Table 6.3.

Despite the small sample sizes, $p < 0.04$ for all but one system evaluated in Table 6.3; this allows us to reject the null hypothesis of no correlation between these systems and the human judgment of similarity. The only exception is the concept definition-based system, for which $p = 0.103$ for the WordSim dataset. This shows that, in the case of OpenCyc where less than half of the concepts have assigned a concept definition, there is no benefit in combining the structure and concept definition-based measures.

Table 6.3: Spearman rank correlations between several systems and the human judgments obtained on three standard datasets (MC, RG and WordSim). The measures marked in *italic* were proposed in this thesis. The measures marked with *(object property)* determine the relatedness between the domain or range of the object property and another concept rather than the object property itself. The best results for the proposed and re-implemented systems are rendered in bold.

Measures used in the evaluation	**MC-OpenCyc Miller and Charles**	**RG-OpenCyc Rubenstein and Good-enough**	**WordSim-OpenCyc Finkelstein et al.**
WeightedConceptPath Log	0.648	0.570	0.373
WeightedConceptPath Log (object property)	0.659	**0.706**	0.390
WeightedConceptPath Sqrt	0.679	0.534	0.399
WeightedConceptPath Sqrt (object property)	**0.691**	0.550	**0.417**
OpenCyc Definition	0.475	0.341	0.195
Moore et al.	0.648	0.559	0.356
Shortest Path Unit Weight	0.587	0.304	0.238
Leacock Chodorow	0.587	0.304	0.238
Wu Palmer	0.552	0.390	0.286

In some cases the concepts in the dataset are mapped to OpenCyc object properties, demanding that we treat object properties different from other types of relations. An example would be the WordNet 3.0 concept *sage* which corresponds to the OpenCyc object property *mentorOf*:

> ***sage*** - a mentor in spiritual and philosophical topics who is renowned for profound wisdom
>
> ***mentorOf*** - (mentorOf PERSON MENTOR) means that MENTOR is the mentor of PERSON, in the sense that MENTOR is a teacher or trusted counselor or advisor of PERSON

In order to determine the shortest path between an object property and a concept we consider the domain and range of the object property. In case the domain and range of the object property are different concepts, we look at both concepts independently and take the shortest weighted path. For example, the domain and range of the *mentorOf* object property is the concept *Person*. The shortest weighted path between *mentorOf* and *Prophet*, using the *WeightedConceptPath Log* measure is: *Person – Teacher – Prophet*. The *WeightedConceptPath Log/Sqrt* object property methods take this observation into account.

For all three standard datasets, the best results are obtained by the proposed structure-based measures of relatedness *WeightedConceptPath Log/Sqrt* which take into account the degree of abstractness of concepts.

6.1.4.2 Experiments on a Subset of OpenCyc Concepts

In this subsection we perform an evaluation on a subset of OpenCyc concepts, and propose a clustering approach for validating the results. The aim is to show that our proposed algorithm relying on weighted concept paths can also be used for clustering concepts based on the similarity between them. In addition, concept weighting and clustering can be useful in applications such as ontology navigation, by showing the user views of the ontology centered around information-rich concepts, as described in (Motta et al., 2011).

We validate the results via the clustering approach described in Section 6.1.2.2. Table 6.4 summarizes the results, showing the modified DBI and the average intra-cluster and inter-cluster distance for each of the proposed algorithms (*WeightedConceptPath Log* and *Sqrt* and *OpenCyc Definition*), as well as of the algorithms we compare against.

Table 6.4: The modified Davies-Bouldin Index (DBI) and the averaged inter-cluster and intra-cluster distances for the dataset comprising pairwise concept distances for a subset of OpenCyc concepts belonging to four different categories. The DBI is used to rank the evaluated algorithms and highlighted in gray. Our proposed algorithms are *WeightedConceptPath Log* and *Sqrt*, and *OpenCyc Definition*, respectively. The best performing algorithms have a low DBI value, low intra-cluster distances and high inter-cluster distances.

Systems used in the evaluation	**Modified Davies Bouldin Index**	**INTRA Cluster Distance**	**INTER Cluster Distance**
WeightedConceptPath Log	**1.363**	0.344	0.564
WeightedConceptPath Sqrt	1.416	0.144	0.233
OpenCyc Definition	1.623	0.582	0.813
Moore et al.	1.408	0.360	0.586
Shortest Path Unit Weight	1.652	0.412	0.597
Leacock Chodorow	1.659	0.225	0.325
Wu Palmer	1.610	0.123	0.162
Random	1.994	0.497	0.508

Using the methods summarized in Table 6.1, we have computed the distance between each two pairs of concepts. The value of the distance between two concepts is lower if the concepts are semantically close, and higher if the concepts are dissimilar. Some algorithms, including our proposed approaches, yield a distance measure between the concepts: *WeightedConceptPath Log* and *Sqrt*, *Moore et al.*, *Shortest Path Unit Weight*. Other algorithms yield a similarity measure: *Leacock and Chodorow*, *Wu and Palmer*, *OpenCyc Definition*. For consistency, the output of the algorithms yielding a similarity measure has been adapted to yield a normalized distance measure (see Section 3.2), allowing an easier comparison among algorithms.

Intuitively, the distance computed between concepts from the same category will be lower than the one between concepts belonging to different categories. Moreover, if we visualize the results, we would expect to identify four different clusters, corresponding to each of the four categories.

The lowest DBI is obtained for the *WeightedConceptPath Log* algorithm, while *WeightedConceptPath Sqrt* and *Moore et al.* also obtain good results. Thus, by differentiating between concept types we can improve the initial distance measure proposed by Rada et al., and outperform other structured and definition-based measures.

For visualizing the results, we use a multidimensional scaling (MDS) approach (Borg & Groenen, 2005). Given the pairwise distances between concepts, MDS assigns each concept a point in the two-dimensional space. Figure 6.3a shows a visualization of concept distances using a purely random measure. As expected, in this visualization, the four clusters are not distinguishable.

As a comparison, we visualize in Figure 6.3b the clustering pattern obtained with the *WeightedConceptPath Log* measure; here we can easily identify the four clusters. The two outlier concepts in the "Fruit" cluster in Figure 6.3b are the OpenCyc concepts *AppleInc* and *Date_ TheProgram*, representing a software and a clock synchronization program, respectively. The algorithm correctly identified them as being closer to the "Computer hardware" and "Computer software" clusters.

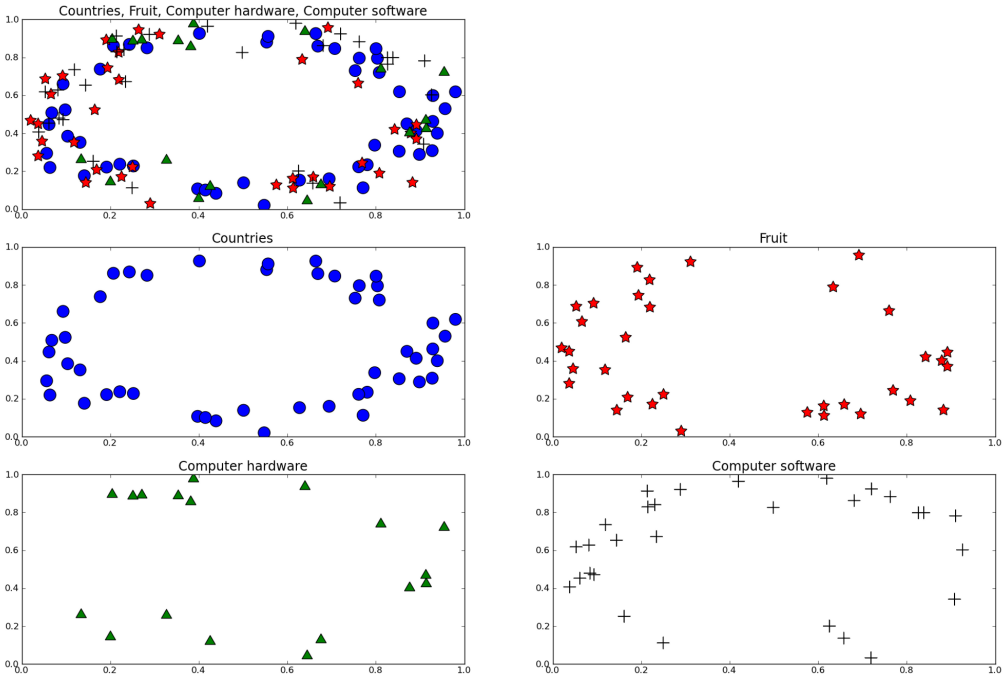
6.1.5 DBpedia

Table 6.5 reports Spearman rank correlations between the human judgment and three algorithms for determining the relatedness between concepts. Due to the fact that only a small number of resources from the three standard datasets have assigned a DBpedia ontology class, we show results when using Wikipedia categories. In order to obtain the relatedness between two DBpedia concepts we start by determining the pairwise relatedness between all categories assigned to the concepts; the final relatedness score between the concepts is given by the maximum relatedness between the corresponding categories.

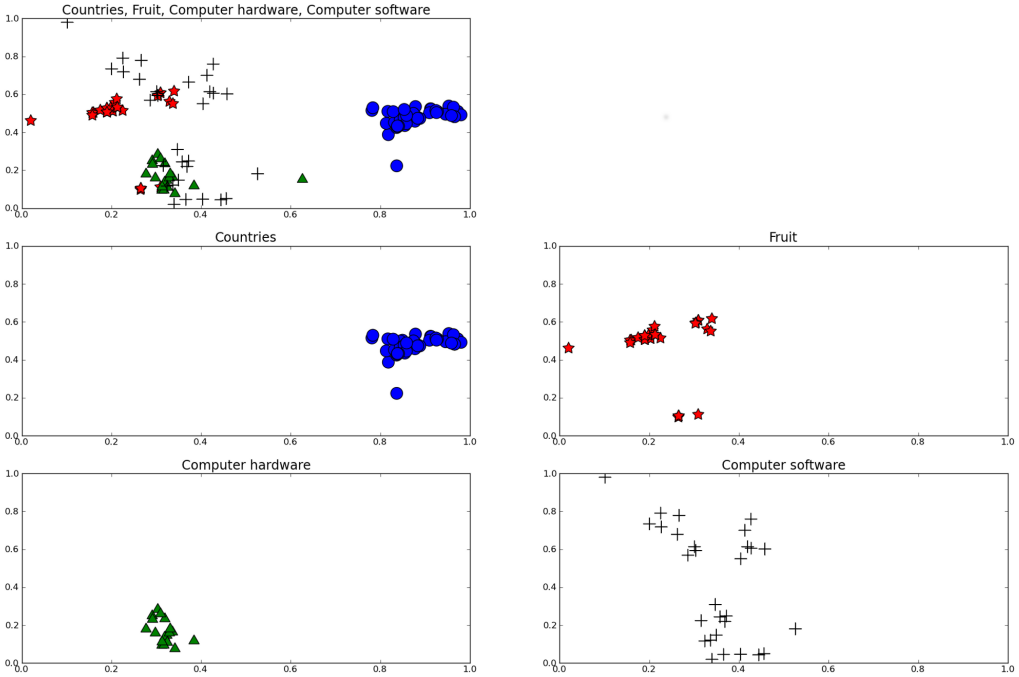
Despite the small sample sizes, $p < 0.003$ for all systems evaluated in Table 6.5, which, as in the case of the WordNet relatedness evaluation, allow to decisively reject the null hypothesis of a system giving a purely random output.

For all three standard datasets, the best results are obtained by combining the concept definitions and the knowledge base structure in the hybrid relatedness measure (*Hybrid Log/Sqrt*), corroborating the results reported for the WordNet experiments.

As in the case of WordNet relatedness evaluation, we make two remarks regarding the concept definition-based relatedness and the hybrid relatedness measures.



(a) *Random* measure.



(b) *WeightedConceptPath Log* measure.

Figure 6.3: A visualization of concept relatedness in the OpenCyc clustering experiment using the (a) *Random* measure and (b) *WeightedConceptPath Log* measure.

Table 6.5: Spearman rank correlations between several systems and the human judgments obtained on three standard datasets (MC, RG and WordSim). The *WeightedConceptPath* measure was described in this thesis and is based on Wikipedia categories. The best results for the proposed and re-implemented systems are rendered in bold.

Measures used in the evaluation	**MC-DBpedia Miller and Charles**	**RG-DBpedia Rubenstein and Good-enough**	**WordSim-DBpedia Finkelstein et al.**
WeightedConceptPath Log (category)	0.841	0.815	0.589
WeightedConceptPath Sqrt (category)	0.819	0.791	0.592
DBpedia Definition	0.879	0.813	0.561
Hybrid Log	**0.920**	**0.876**	0.641
Hybrid Sqrt	0.913	0.865	**0.650**
Moore et al. (category)	0.843	0.815	0.464
Shortest Path Unit Weight (category)	0.815	0.790	0.421
Adapted Google Distance	0.586	0.493	0.527

Concept definition-based relatedness. In order to determine the contribution of connected concepts to the concept definition-based measure we use the same evaluation settings described for WordNet, namely we assign a weight $\alpha = 1$ to the term vector corresponding to the definition of the concept itself, and vary the weight of the term vectors belonging to connected concepts. Figure 6.4 shows Spearman rank correlations when varying the definition weight α.

Results show that connected concepts are useful for improving the correlation between the *WordNet Definition* system and the human judgment, though to a lesser extent than for WordNet. This is because the DBpedia definitions which are formed of DBpedia short or long abstracts describe the concept in more detail compared to the short WordNet glosses. As for WordNet, the performance of the concept definition-based system can be further improved by weighting the contribution of connected concepts, especially in the case of the WordSim dataset. We report *DBpedia Definition* results for $\alpha = 0.5$ in Table 6.5.

The Hybrid measure of relatedness. For determining the value of the ζ hybrid weight we conduct similar experiments as in the case of WordNet. Figure 6.5 provides an overview of our findings by depicting different Spearman rank correlations depending on the hybrid weight ζ. As the *DBpedia Definition* measure slightly outperforms the structured-based measures *WeightedConceptPath Log/Sqrt* on the Miller and Charles and Rubenstein and Goodenough datasets, we obtain best results for $\zeta = 0.8$. However, for the WordSim dataset correlation improves for a lower value of ζ. Because WordSim is the larger dataset covering more concept pairs, we report results for $\zeta = 0.3$ in the Table 6.5 table for the *Hybrid* measure.

For the DBpedia knowledge base, information provided by both the structure and the concept definitions yields the best rank correlations with human judgment.

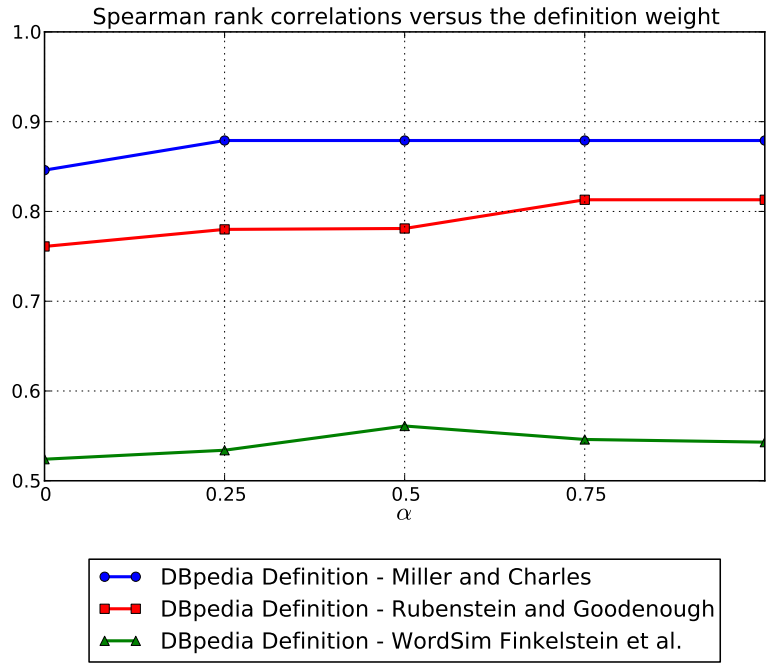


Figure 6.4: Spearman rank correlations for varying definition weight α for DBpedia concepts. The results are obtained using the three standard datasets described in the evaluation settings.

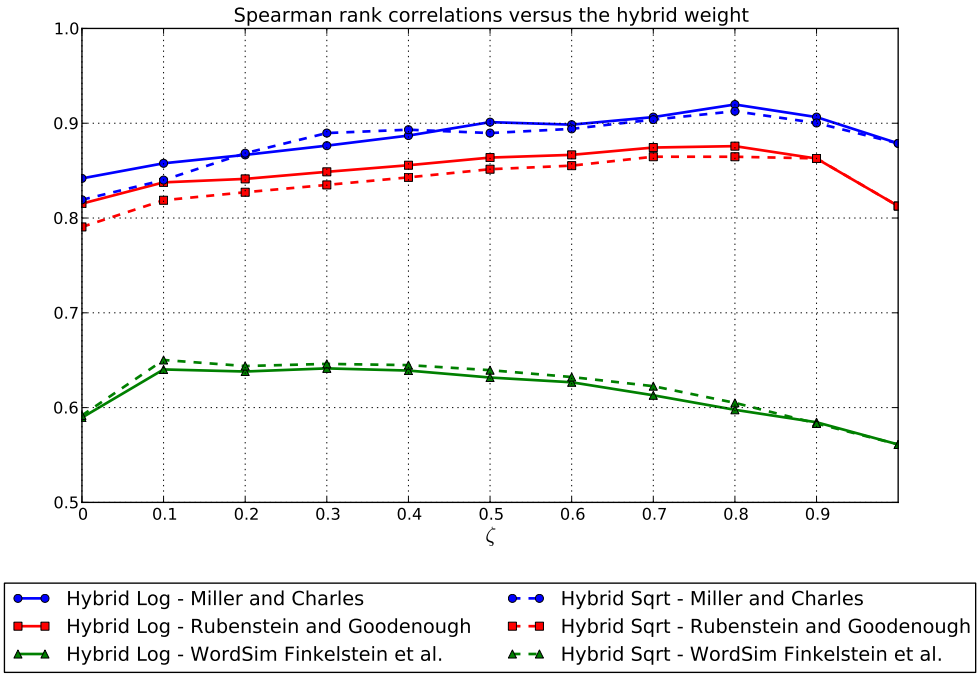


Figure 6.5: Spearman rank correlations for varying hybrid weight ζ for DBpedia concepts. The results of the *Hybrid Log* and *Sqrt* measures are obtained using the three standard datasets described in the evaluation settings.

6.2 Text Annotation

In this section we evaluate the performance of the proposed text annotation framework and present experiments for two knowledge bases: WordNet and DBpedia, for which evaluation datasets are readily available. In the following we describe two evaluation datasets which we use in our experiments as well as the evaluation metrics and the results obtained for each knowledge base.

For each knowledge base we show different configurations of the proposed text annotation framework, by varying the concept relatedness measure, the function for aggregating relatedness scores and the local context window size.

As both *WeightedConceptPath - Log* and *WeightedConceptPath - Sqrt* yield similar results in the relatedness evaluation experiments we only use one of the measures, namely *WeightedConceptPath - Log* for the text annotation evaluation; henceforward we are going to refer to this measure as *WeightedConceptPath*. Following the relatedness evaluation results, the term weight α for the concept definition-based relatedness measures *WordNet Definition* and *DBpedia Definition*, respectively, is set to 1 in the case of the concept itself and 0.5 in the case of related concepts. We set the *hybrid weight* $\zeta = 0.6$ for the WordNet text annotation evaluation experiments and $\zeta = 0.3$ for the DBpedia experiments as for these value of ζ we obtained good results in the relatedness evaluation (see Section 6.1).

6.2.1 Evaluation Dataset Description

WordNet annotations. We evaluate text annotation based on WordNet using a dataset proposed in the SemEval 2010 workshop, Task 17 (Agirre et al., 2010), which comprises corpora from the environment domain. This is a multilingual task for Chinese, Dutch, English and Italian. In this work we focus on English; however, our approach is language independent and can be applied to the other languages as well, provided the availability of the WordNet ontology for the specific language. The English dataset contains three texts with 1,032 nouns and 366 verbs to be annotated with WordNet concepts. Additionally, the workshop organizers provide 113 background documents on related subjects which can be used for training.

DBpedia annotations. They are evaluated on the dataset provided by the SemEval 2013 workshop, Task 12 (Navigli et al., 2013). This dataset consists of 13 documents spanning different domains such as finance, politics or sports in 5 languages: English, French, German, Italian and Spanish. Participating systems are required to provide either BabelNet annotations or, alternatively, WordNet or Wikipedia annotations. The English dataset to be annotated with Wikipedia concepts comprises 1242 noun instances, out of which 945 are single-words, 102 are multi-word expressions and 195 are named entities.

We make use of this dataset and automatically map the English Wikipedia annotations represented as Wikipedia article titles to DBpedia 3.2 resources. This is straightforward as each DBpedia resource URI is derived from the corresponding Wikipedia article URL (see Section 4.3). As we work with an older version of the DBpedia knowledge base, the mapping results in 1220 nouns linked to DBpedia 3.2 concepts (for 22 Wikipedia articles we did not identify a corresponding DBpedia 3.2 concept); 163 named entities have a corresponding DBpedia ontology class.

6.2.2 Evaluation Metrics

We evaluate the systems in terms of standard evaluation metrics used by the semantic evaluation workshops: *precision*, *recall* and *F-measure*.

Precision represents the fraction of concept annotations generated by a system which are equivalent to the golden standard ones:

$$Precision = \frac{|\{correct\ annotations\}| \cap |\{retrieved\ annotations\}|}{|\{retrieved\ annotations\}|} \tag{6.5}$$

Recall represents the fraction of correct concept annotations which the system generates:

$$Recall = \frac{|\{correct\ annotations\}| \cap |\{retrieved\ annotations\}|}{|\{correct\ annotations\}|} \tag{6.6}$$

The aim is to build a text annotation system which exhibits high precision, i.e. the concepts suggested as annotations for the words and collocations in the text fragment match the golden standard ones, and high recall, i.e. the system generates annotations for as many words or collocations in the text fragment as possible, and these annotations match the golden standard ones.

F-measure represents the harmonic mean of precision and recall:

$$F - measure = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{6.7}$$

In order to validate that one system X significantly outperforms another system Y we want to reject the null hypothesis H_0: "X performs worse or equal to Y", using the following test statistic t:

$$t(o_X, o_Y) = |e(o_X) - e(o_Y)| \tag{6.8}$$

The distribution of t under the marginal case of the null hypothesis can be sampled using an approximate randomization technique (Noreen, 1989; Yeh, 2000). This involves flipping the annotations given by the two systems independently for each word with a probability of 0.5.

6.2.3 WordNet

Figure 6.6 shows the annotation results (F-measure) obtained for this dataset, for all words. As our algorithm identifies candidate concepts for all words to be annotated, yielding annotations for all words, precision and recall are equal.

Based on the Spearman rank correlation results we use two relatedness measures to test the annotation framework: *WeightedConceptPath* and *WordNet Definition*, as well as the *Hybrid* measure which is a weighted combination of the two. We experiment with different settings for our text annotation framework:

- **Concept relatedness measure**. The *WeightedConceptPath* and *WordNet Definition* measures use different information to determine the relatedness between concepts: the first one relies on the WordNet concept graph while the latter one is based on concept definitions. The *WeightedConceptPath* measure outperforms the *WordNet Definition* one on nouns over all window sizes

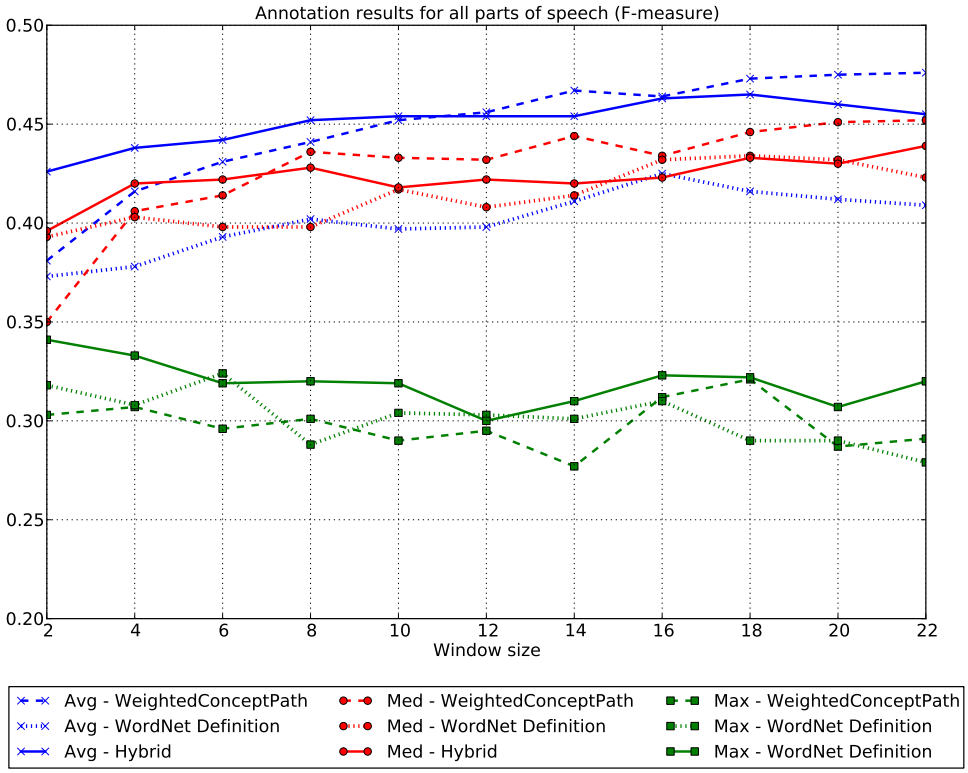
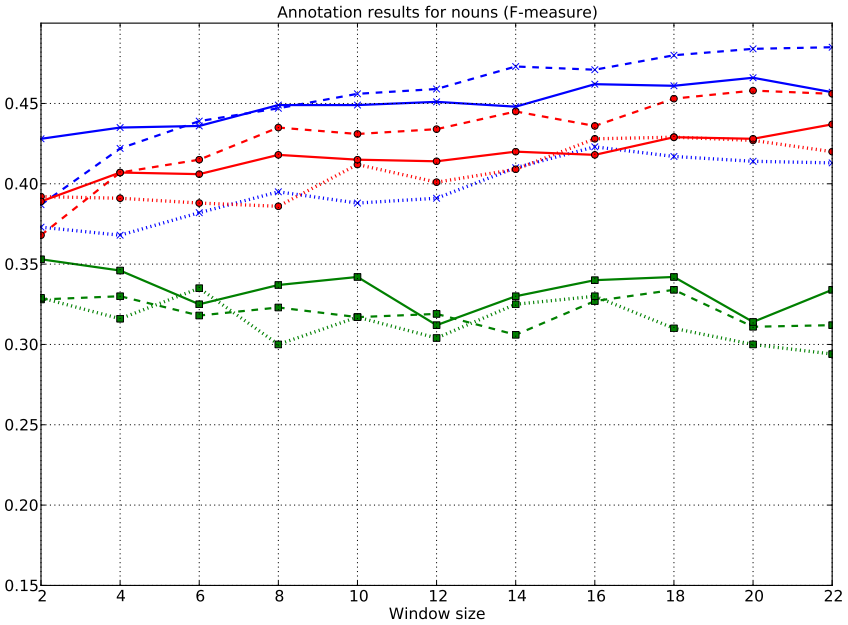


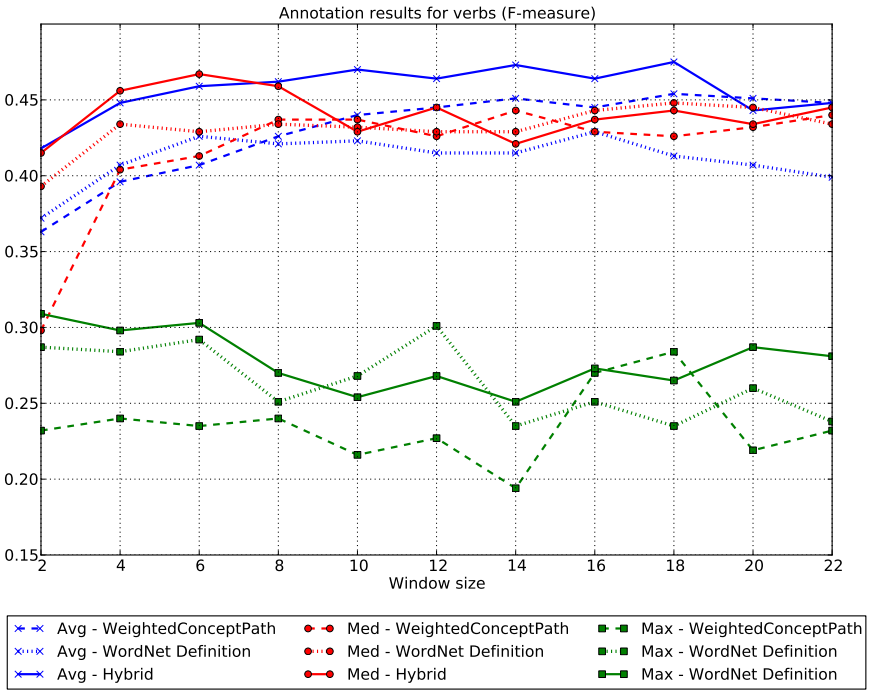
Figure 6.6: Annotation results (F-measure) obtained for all words, using the SemEval 2010 Task 17 dataset. The *WeightedConceptPath*, *WordNet Definition* and *Hybrid* relatedness measures and *average*, *maximum* and *median* aggregate functions were used in the experiments.

(see Figure 6.7a). Annotating verbs is a more difficult task, also due to the higher number of candidate concepts per word compared to nouns, and the fine grained differences between these concepts. In the case of verbs concept definitions are more useful for the annotation task when a small window size is considered (see Figure 6.7b). Moreover, the annotation system based on the *Hybrid* measure outperforms the systems based on the *WeightedConceptPath* and *WordNet Definition* measures in the case of verbs and when using few words from the local context.

- **Aggregate function**. We experiment with three such functions: *maximum*, *average* and *median*. The results of the algorithms which aggregate the relatedness score using the *maximum* function are highly dependent on the size of the context window. This is due to the fact that the candidate concept with the maximum relatedness to the local context is chosen for annotation, which can differ significantly as more concepts are included in the local context. This is not the case with the *average* or *median* functions, where all relatedness results for a candidate concept are taken into account. Moreover, results based on the *maximum* function are worse compared to those of the other two aggregate functions.



(a) Annotation results for nouns.



(b) Annotation results for verbs.

Figure 6.7: Annotation results (F-measure) obtained for nouns and verbs individually, using the SemEval 2010 Task 17 dataset. The *WeightedConceptPath*, *WordNet Definition* and *Hybrid* relatedness measures and *average*, *maximum* and *median* aggregate functions were used in the experiments.

- **Local context window sizes**. We test for multiple window sizes in an incremental manner. In the case of the *average* or *median* functions results improve as more words are included in the local context. On the other hand, the *maximum* function does not exhibit the same improvement, as larger contexts bring about more noise.

We obtain highly significant results when comparing the *WeightedConceptPath average* with the *WordNet Definition average*, or the *WeightedConceptPath median* with the *WordNet Definition median* algorithms, respectively; in both cases $p < 10^{-4}$, for window size 22. We could not reject the null hypothesis for the *WeightedConceptPath max* and *WordNet Definition max* algorithms on any reasonable significance level; in this case we obtained $p = 0.37$.

Our algorithm compares well with other SemEval 2010 systems participating in this task (see Table 6.6). All of the best performing knowledge based systems (A. Kulkarni et al., 2010; Tran et al., 2010; Soroa et al., 2010) make use of domain-specific corpora to construct the knowledge base and select only those candidate concepts that belong to this domain-specific knowledge base. The framework that we propose does not require additional corpora; the annotation algorithm relies only on the local context information available in the input documents and the ontology used as concept inventory. Yet even without additional domain-specific corpora, our algorithm performance is comparable to the best knowledge based systems participating in the evaluation workshop.

Table 6.6: Annotation results of the best knowledge-based approaches participating in the SemEval 2010 Task 17 workshop, ordered by recall, as provided in Agirre et al., 2010. Notice that all systems use domain-specific corpora (marked with *DS*) while our system is domain independent and does not use external resources beyond the ontology.

System	**Type**	**Precision**	**Recall**	**Recall Nouns**	**Recall Verbs**
Most frequent sense	-	0.505	0.505 ± 0.023	0.519 ± 0.026	0.464 ± 0.043
CFILT-3	DS	**0.512**	**0.495 ± 0.023**	**0.516 ± 0.027**	0.434 ± 0.048
Treematch	DS	0.506	0.493 ± 0.021	**0.516 ± 0.028**	0.426 ± 0.046
Treematch-2	DS	0.504	0.491 ± 0.021	0.515 ± 0.030	0.425 ± 0.044
kyoto-2	DS	0.481	0.481 ± 0.022	0.487 ± 0.025	0.462 ± 0.039
Treematch-3	DS	0.492	0.479 ± 0.022	0.494 ± 0.028	0.434 ± 0.039
Our System	O	0.476	0.476	0.485	0.448
RACAI-MFS	DS	0.461	0.460 ± 0.022	0.458 ± 0.025	**0.464 ± 0.046**
UCF-WS	DS	0.447	0.441 ± 0.022	0.440 ± 0.025	0.445 ± 0.043
HIT-CIR-DMFS-1.ans	DS	0.436	0.435 ± 0.023	0.428 ± 0.027	0.454 ± 0.043
UCF-WS-domain	DS	0.440	0.434 ± 0.024	0.434 ± 0.029	0.434 ± 0.044
IIITH2-d.r.l.baseline.05	DS	0.496	0.433 ± 0.024	0.452 ± 0.023	0.390 ± 0.044

6.2.4 DBpedia

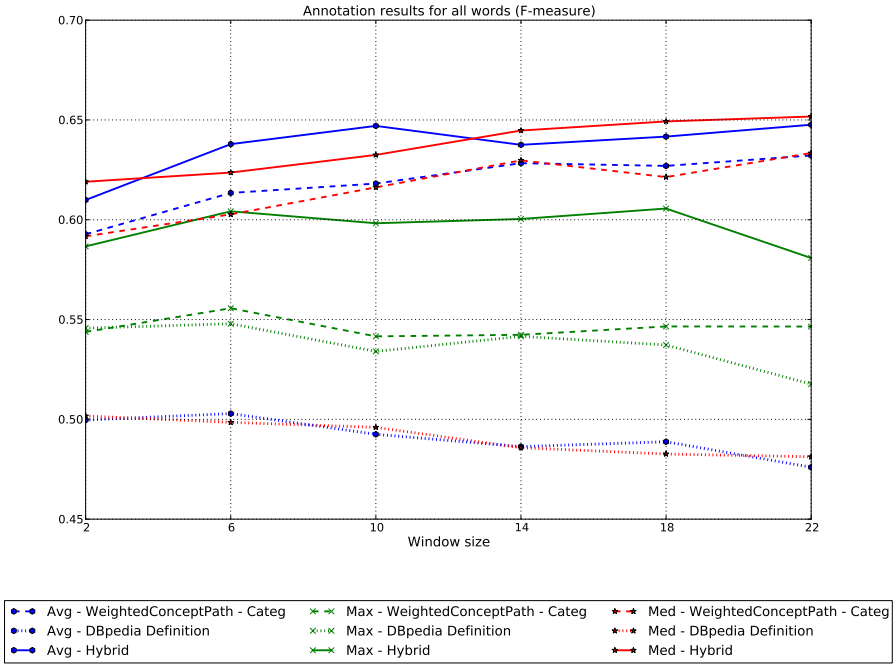
Figure 6.8 depicts the annotation results (F-measure) for the SemEval 2013 Task 12 dataset, when we attempt to annotate all words. Because the DBpedia ontology concepts do not cover the entire dataset, but rather only named entities, we make use of the Wikipedia categories. For each candidate concept we retrieve all its categories by following *dcterms:subject* links. Next, we determine the relatedness between two concepts via their categories, by computing the pairwise relatedness between all categories and keeping the maximum value. Our assumption is that if two concepts are related, then there will be at least a pair of categories belonging to the two concepts which are also related. We use the category graph G_{Dk} to determine the shortest weighted path.

As in the case of the WordNet annotation experiments, we discuss several settings:

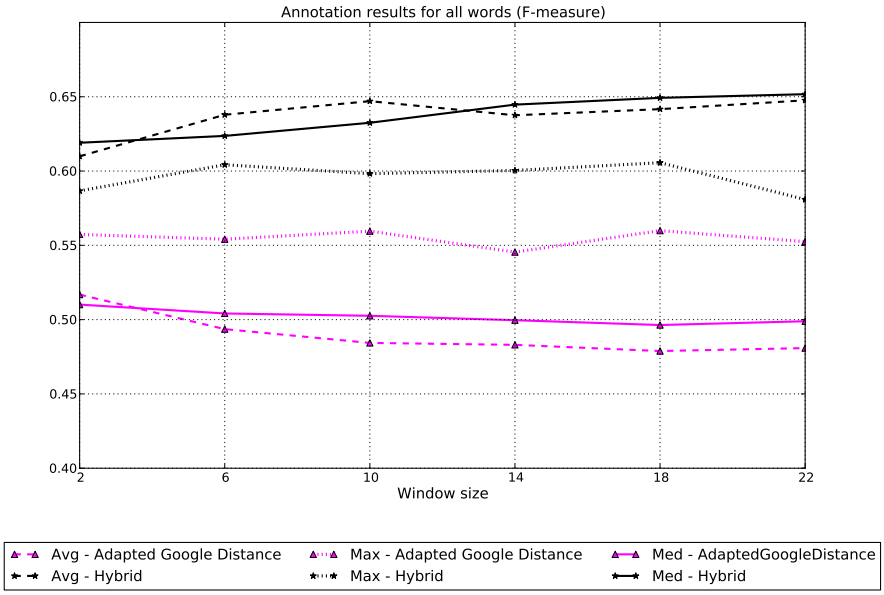
- **Concept relatedness measure**. We use four relatedness measures to test the annotation framework: *WeightedConceptPath (category)*, which takes into account Wikipedia categories for determining the relatedness between concepts, *DBpedia Definition*, the *Hybrid* measure as a combination of the two aforementioned measures and *Adapted Google Distance*. *DBpedia Definition* and *Adapted Google Distance* both use the entire DBpedia knowledge base for determining the relatedness between concepts. The *Adapted Google Distance* measure is not defined for unrelated concepts which are more than two steps apart. The *DBpedia Definition* measure quantifies the degree of relatedness between two concepts based on the similarity of their textual descriptions. Yet two related concepts might not be described using the same words. The best results are obtained when combining the *WeightedConceptPath* using the category subgraph and *DBpedia Definition* measures into the *Hybrid* measure. The advantage of the *WeightedConceptPath* measure is that it determines weighted paths for concepts that are connected via an arbitrary number of steps while penalizing paths that include more abstract categories. On the other hand, the *DBpedia Definition* measure which takes into account DBpedia abstracts has a good performance in the case of concepts that are related.

- **Aggregate function**. We experiment with three such functions: *maximum*, *average* and *median*. The observation we made for WordNet regarding the dependence of algorithm results on the size of the context window is valid here as well; the results of algorithms which aggregate relatedness scores using the *maximum* function depend more on the size of the context window compared to results of algorithms which use the other two functions. Similar to WordNet, the best results are obtained by algorithms implementing the *WeightedConceptPath (category)* measure and aggregating all relatedness results for a candidate concept with the *average* or *median* functions. On the other hand, algorithms relying on the *Adapted Google Distance* or *DBpedia Definition* relatedness measures performed worse when the relatedness scores are aggregated via the *average* or *median* functions. This is due to the fact that both measures are useful for identifying concepts which are related but have problems with unrelated concepts.

- **Local context window sizes**. Similar to WordNet, we test for multiple window sizes in an incremental manner. Results improve with the increase in



(a) Annotation results using the *WeightedConceptPath*, *DBpedia Definition* and *Hybrid* relatedness measures.



(b) Annotation results using the *Hybrid* and *Adapted Google Distance* relatedness measures.

Figure 6.8: Annotation results (F-measure) obtained for all words, using the SemEval 2013 Task 12 dataset. The *WeightedConceptPath*, *DBpedia Definition*, *Hybrid* and *Adapted Google Distance* relatedness measures and *average*, *maximum* and *median* aggregate functions were used in the experiments.

window size for the *WeightedConceptPath* measure while the opposite happens for the other two measures. The reason is that by adding more context words we also increase the number of unrelated concepts.

We use the same procedure described for WordNet to validate the DBpedia results. We obtain highly significant results when comparing the following pairs of algorithms: *WeightedConceptPath average* using the category graph with the *DBpedia Definition average* and *Adapted Google Distance average*, or *WeightedConceptPath median* using the category graph with the *DBpedia Definition median* and *Adapted Google Distance median*. For all the aforementioned cases $p < 10^{-4}$, for window size 22.

Next, we conduct experiments for named entities only, and report results in Figure 6.9. We want to know if annotations obtained via the class graph outperform the ones provided by using the category graph. Even in the case of named entities category information turns out to be more useful than using the DBpedia ontology class hierarchy. We invoke two reasons: a) the relative small size of the DBpedia ontology that we used in the experiments and b) the fact that this ontology forms a shallow subsumption hierarchy where the average depth of leaf classes is only 2.4 (Paulheim & Bizer, 2013).

Future work should investigate the latest version of the ontology, which includes a much richer subsumption hierarchy and more relations between classes.

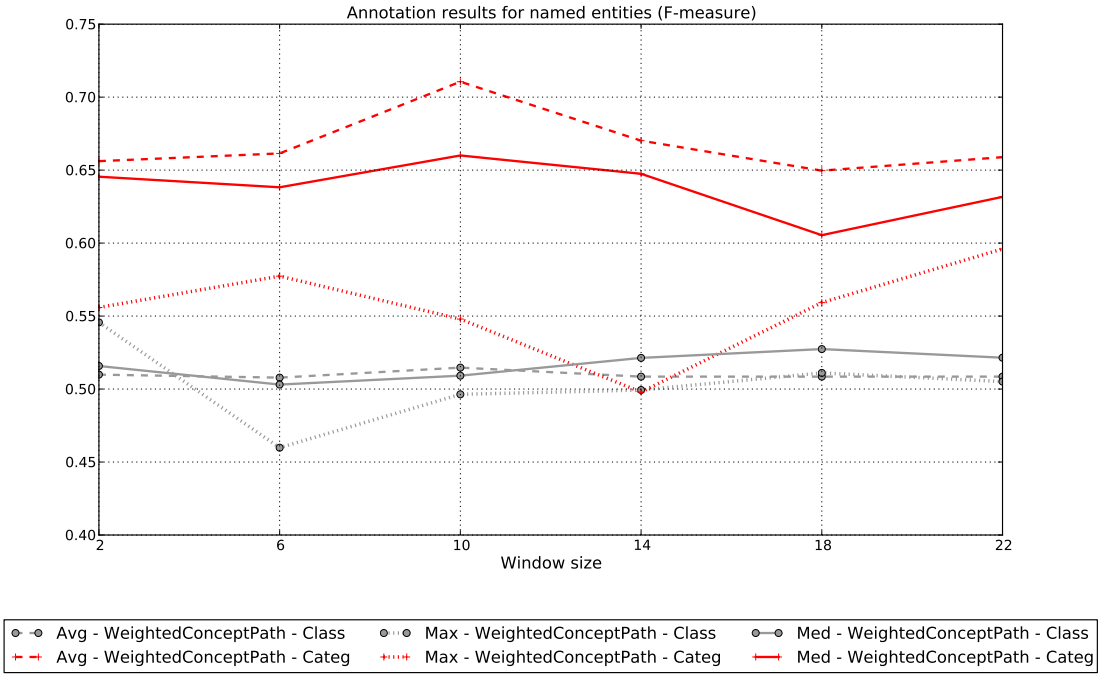


Figure 6.9: Annotation results (F-measure) obtained for named entities, using the SemEval 2010 Task 17 dataset. The plot shows the *WeightedConceptPath* measure using the class and category graphs. The *average*, *maximum* and *median* aggregate functions were used in the experiments.

Even though we cannot directly compare our results with the SemEval 2013, Task 12 results because of using a different concept inventory, we note that only one system provided Wikipedia-based annotations. This system uses a version of the Personalized Page Rank algorithm which incorporates concept frequencies (Gutierrez et al., 2013). The authors submitted three versions of the system, differentiated by the initialization of the ranking algorithm with a set of seeds. These seeds are either candidate concepts corresponding to all nouns in the sentence, all words in the sentence or all nouns in the document. Their best result was 0.622 precision and 0.489 recall, yielding a 0.548 F-measure. Their named entity F-measure score was 0.864. However, the disambiguation is performed on an extended version of the WordNet knowledge base rather than on BabelNet, and only in the final step WordNet synsets are assigned the corresponding BabelNet synsets and Wikipedia pages.

Our framework is more general in that it makes use of the DBpedia ontology and knowledge base to perform text annotation and does not rely on the existing links between WordNet and DBpedia.

Table 6.7 summarizes the best annotation results obtained by the proposed text annotation framework.

Table 6.7: The best annotation results of the proposed text annotation framework on the SemEval 2013 Task 12 dataset, when using DBpedia as a concept inventory.

System	**Precision**	**Recall**	**F-measure**
WeightedConceptPath (category) Avg	0.644	0.623	0.633
DBpedia Abstract Max	0.552	0.544	0.548
Hybrid Med	**0.662**	**0.641**	**0.652**
Adapted Google Distance Max	0.564	0.555	0.560
WeightedConceptPath (category) Avg named entities	**0.711**	**0.711**	**0.711**
WeightedConceptPath (class) Max named entities	0.546	0.546	0.546
DBpedia Abstract Med named entities	0.655	0.655	0.655
Adapted Google Distance Max named entities	0.404	0.404	0.404

6.3 Summary

We conclude this chapter by making several remarks based on the results obtained in the evaluation settings.

Regarding relatedness measures, in the case of WordNet and DBpedia, approaches based on both the knowledge base structure and the concept definitions yield the best rank correlations with human judgment, showing that the knowledge base structure and concept definitions act as two sources of complementary information. For OpenCyc where less than half of the concepts have assigned a concept definition, there is no benefit in combining the structure and concept definition-based measures.

The WordNet and DBpedia-based text annotation experiments show that our annotation framework, even if relying only on the information provided by the knowledge base, yields competitive results.

The following chapter discusses these remarks in more detail.

Chapter 7

Discussion

In this chapter we discuss the evaluation results for the proposed relatedness measures as well as the text annotation framework as a whole.

7.1 Relatedness Measures

In this thesis we proposed different types of relatedness measures: a concept definition-based measure which builds a Vector Space Model of concept definitions, a structure-based measure which is based on determining shortest weighted paths between concepts and a hybrid measure which is a weighted combination of the aforementioned two measures. The measures do not require additional corpora aside from the ontology or knowledge base itself. This is an important feature, as we showed that acquiring information from additional corpora is expensive and domain dependent. We apply the measures in the case of three knowledge bases exhibiting different characteristics: WordNet, OpenCyc and DBpedia. WordNet is a lexical database which is mainly organized around specific concepts called synsets. OpenCyc is a general-purpose ontology with several abstract concepts for grouping information. DBpedia consists of a large number of specific concepts classified in a shallow ontology, where each concept corresponds to a Wikipedia article.

The concept definition-based measure proposed in this thesis can be seen as an extension of the work described by Patwardhan (2003). We determine the relatedness between two concepts by taking into account the definition of each concept as well as the definitions of connected concepts. Instead of treating all concepts as equally relevant for the final relatedness score, as proposed in (Patwardhan, 2003), our approach is more general as it allows differentiating between concepts via *definition weights*. These weights are assigned to term vectors corresponding to concept definitions (see Section 3.1.1).

Our relatedness measure tends to perform well on all three standard datasets when using WordNet or DBpedia as reference knowledge bases. However, the same results are not reproducible in the case of OpenCyc. WordNet concepts are assigned a gloss which is a short textual description, and in some cases example sentences while DBpedia concepts have either a short or a long abstract extracted from the Wikipedia page text content. On the other hand, less than half of OpenCyc concepts have assigned a definition.

For concept pairs where humans assign a high relatedness score, this type of measure exhibits high correlation with human judgment. For example, both human judgment and the definition-based measure assign a high relatedness score to concept

pairs such as *(coast - shore)* or *(football - soccer)*. However, the definition-based measure has low sensitivity for concept pairs where humans assign a low relatedness score. This is the case of concept pairs such as *(noon - string)* or *(chord - smile)* which are assigned a score of 0.8 and 0.13, respectively, out of 4 by the human assessors and 0 by the definition-based measure (based on the WordNet knowledge base). We use the term *sensitivity* of a relatedness measure to describe the ability of that measure to detect small degrees of relatedness.

Structure-based measures are based on assigning weights to knowledge base concepts and effectively aggregating these weights. The goal is to be able to distinguish between concepts depending on their degree of abstractness: more abstract or general concepts with a higher number of relations and more specific concepts with a lower number of relations.

The proposed relatedness measures which take advantage of the knowledge base structure have good performance for all three knowledge bases under consideration, namely WordNet, OpenCyc and DBpedia, on both standard datasets and synthetic data (in the case of OpenCyc), indicating the robustness of the approach. Moreover, this type of measure has a higher sensitivity compared to the definition-based measure. For example, the concept pairs *(noon - string)* or *(chord - smile)* are both assigned low relatedness scores different from 0 by the structure-based measure (based on the WordNet knowledge base), as in both cases there is a weighted path connecting the concepts.

In general, approaches that use unit weighting in determining the shortest path (by counting the number of edges between two concepts) are outperformed by approaches that employ a weighting scheme based on the knowledge base characteristics. As the comparison in Figure 7.1 shows, the unit weight shortest paths have a smaller number of edges than the shortest paths obtained using other weighting schemes, such as the node degree.

On average, the maximum degree of nodes on the unit weight shortest paths is higher than the one on paths obtained using *WeightedConceptPath Log* weights (see Figure 7.2). Therefore the unit weight shortest paths are less informative, as they contain more abstract nodes with higher node degrees. Figure 7.1 and Figure 7.2 graphically depict these observations, using OpenCyc as the underlying knowledge base.

The OpenCyc knowledge base construction explains some of the disagreement with human judgment of relatedness:

- Some concepts are not connected in OpenCyc. For example *Midday* is a subclass of *QualitativeTimeOfDay*, but there is no connection to *TimeOfDay*. This results in a weak connection between *Midday* and *TimeOfDay_NoonHour* even if the human judgments rate the pair among the most related.

- There exist concepts which are connected via few relationships, and for which humans assign a lower relatedness score. There are several such cases, e.g. the word pair "cell - phone" corresponds to the OpenCyc concepts *(CellularTelephone – Telephone)* and was rated with a score of 7.81 out of 10 by the human assessors or the word pair "tiger-cat" corresponding to the OpenCyc concepts *(Tiger – FelidaeFamily)*, which got a 7.35 score.

- There exist concepts that are connected via abstract concepts (with high node degree), e.g. the pair *(DividendPaymentObligation – Paying)* is connected via

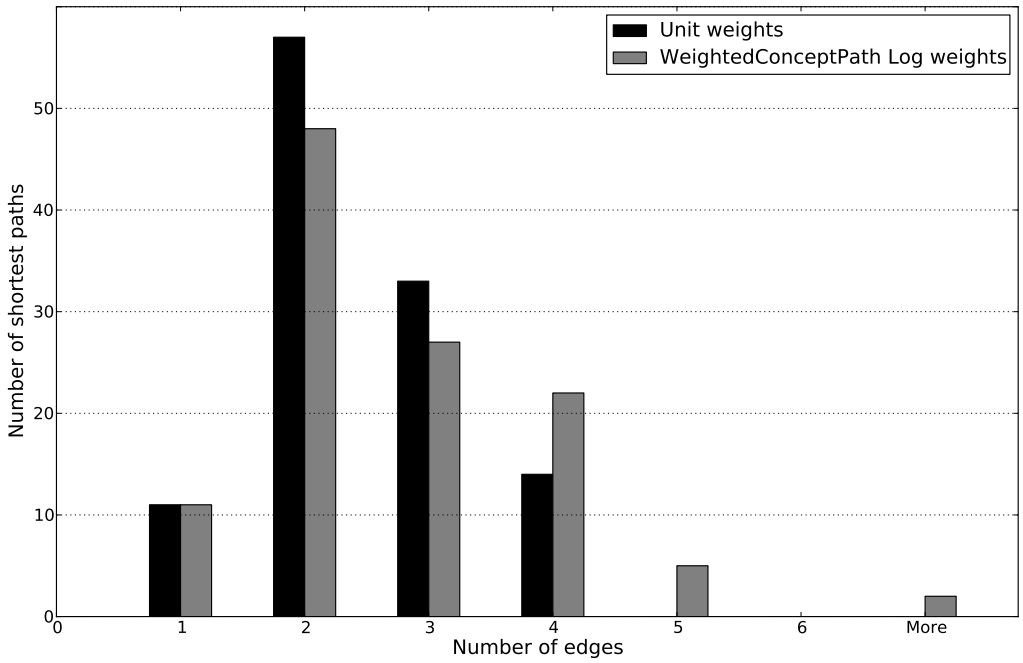


Figure 7.1: The number of edges in OpenCyc shortest paths, using the unit weights and *WeightedConceptPath Log* weights.

CulturalActivity, *TemporalStuffType*, the latter having a node degree of $2,567$. Human assessors rated this pair with a 7.63 score.

The Hybrid measure is a weighted sum of the definition-based measure and the structure-based measure, respectively. This relatedness measure had the best performance for both WordNet and DBpedia knowledge bases on all three standard datasets. By combining the two types of relatedness measures, i.e. definition-based and structure-based, the resulting hybrid measure has higher sensitivity for concept pairs where humans assign a low relatedness score (as we take into account the knowledge base structure) while at the same time obtains high correlation with human judgment for concept pairs where humans assign a high relatedness score (by taking into account concept definitions).

7.2 Text Annotation

In this thesis we proposed a modular yet generic text annotation framework which can be applied to assign concepts to words in a text fragment using different knowledge bases as input. Rather than taking into account specific characteristics of a particular knowledge base we aim to generalize across different knowledge bases. Moreover, we do not make use of additional corpora aside from the knowledge base itself. We select two popular concept inventories, namely WordNet and DBpedia, and show that our framework provides competitive results for both cases. However, there are a number of challenges when using different knowledge bases for text annotation, which are highlighted in the experimental evaluation.

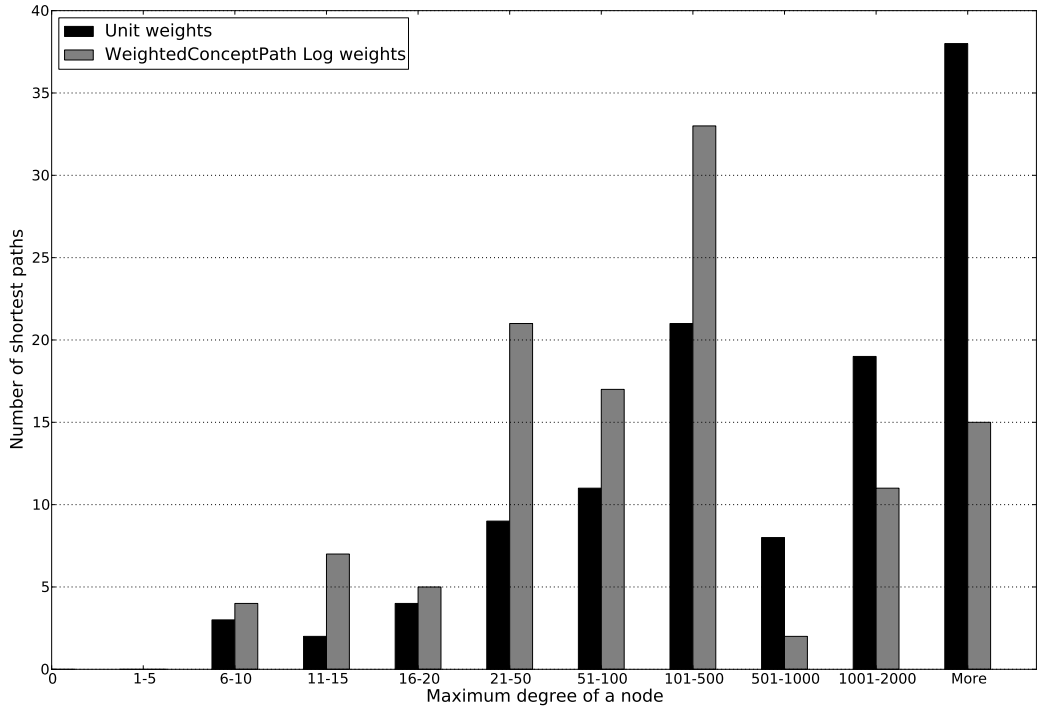


Figure 7.2: The maximum degree of nodes in OpenCyc shortest paths, using unit weights and *WeightedConceptPath Log* weights.

Firstly, the text annotation system integrating the structured-based relatedness measure has a better overall performance compared to the system which integrates the definition-based measure for both WordNet and DBpedia evaluations. The text annotation algorithm ranks the candidate concepts of a word or collocation based on the relatedness score between each candidate concept and the local context, with most of these relatedness scores being low. The structure-based measure takes into account the entire knowledge base graph, whereas the definition-based measure uses only the information provided by the concepts in the local context, exhibiting low sensitivity for concept pairs where humans would assign a low relatedness score. This low sensitivity of the definition-based measure affects the ranking algorithm results, especially in the case of DBpedia.

Even if the concept definition-based measure tends to perform well on the WordNet relatedness evaluation dataset, on the text annotation task the knowledge base structure provides to be more useful for annotating nouns and verbs alike. In the case of the DBpedia evaluation, the method that takes advantage of the category hierarchy outperforms the concept definition-based measure on the text annotation task even if both measures have comparable performance on the relatedness task. Additionally, weighted concept paths determined on the DBpedia category subgraph turn out to be useful for annotating all words in context including named entities.

Annotation results obtained when using the hybrid relatedness measure depend on the input knowledge base. For the WordNet evaluation, overall results are not improved by integrating the hybrid measure in the text annotation framework, as in this case the annotation systems relying on concept definitions and knowledge

base structure, respectively, obtain comparable results. The opposite is true for the DBpedia evaluation, where the results obtained by the text annotation system integrating the structure-based relatedness measure are improved by adding concept definition information.

Secondly, the way relatedness measure results are aggregated across the concepts in the local context depends on the knowledge base used as input and the relatedness measure. In the case of WordNet evaluation, aggregate functions such as *average* or *median* that consider the contribution of all the concepts in the local context outperform functions such as *maximum* which select only one concept. A similar outcome is observed for DBpedia evaluation when using the category hierarchy. The opposite is true for the *Adapted Google Distance* and *DBpedia Definition* measures when applied to DBpedia; in this case the best results were obtained by identifying the concept which is most related to any of the concepts in the local context. The reason is that candidate concepts for a given word in DBpedia span across different categories. In a previous example (see Section 5.2.4) the word *ministers* could be assigned with a Christian minister, a politician, a diplomat or a satirical British sitcom. As the concept denoting a sitcom is very different from the concept denoting a politician, by including the contribution of all candidate concepts, some of them highly related and others completely unrelated, the importance of the highly related concepts diminishes. The *WeightedConceptPath* measure can capture these differences to a higher extent compared to the *Adapted Google Distance* or *DBpedia Definition* measures. WordNet, on the other hand, contains many similar candidate concepts for a given word; some are similar to the point that it is hard even for a human observer to clearly mark the difference.

Finally, the choice of the local context window size is related to both the input knowledge base and the relatedness measure. It is beneficial to use wider local contexts provided robust enough relatedness and aggregation methods. However, wider local contexts also imply higher computational complexity. For WordNet results improve with an increasing window size for both structure and concept definition-based measures when considering the contribution of all concepts in the local context. In the case of DBpedia this is only true for the *WeightedConceptPath* measure, while the performance of the *Adapted Google Distance* and *DBpedia Definition* measures decreases with a wider window size if the contribution of all concepts is taken into account. When aggregating relatedness scores using the *maximum* function, the size of the local context does not influence annotation performance to a great extent, regardless of the relatedness measure.

As a general conclusion to this chapter, evaluation performed on different knowledge bases shows that text annotation results are highly dependent on the quality and coverage of the knowledge base.

Chapter 8

Conclusions

In this thesis we addressed the problem of automatically annotating text with concepts defined in background knowledge datasets, and relying on concept relatedness measures.

Our analysis presented a number of drawbacks of the relatedness measures proposed so far. First, existing concept definition-based approaches which use a Vector Space Model treat all concept definitions in a uniform manner. Second, existing structure-based relatedness measures do not distinguish between the types of concepts which can appear in an ontology or knowledge base. Starting from these observations we proposed a) a more general concept definition-based measure of relatedness which weights the contribution of different concept definitions and b) a structure-based measure relying on a concept weighting scheme applicable to ontologies and knowledge bases where the distances between more specific concepts and the distances between more abstract concepts do not have the same interpretation. The structure and definition-based measures were combined in a hybrid measure of relatedness.

The proposed concept relatedness measures were integrated in a generic text annotation framework for linking text with concepts defined in background knowledge datasets. The modularity of the framework allowed us to experiment with various settings, assessing the influence of different relatedness measures, of the aggregation functions involved in the ranking of candidate concepts and of the local context window size.

The evaluation settings highlighted the advantages and shortcomings of these approaches and presented results for ontologies and knowledge bases with different characteristics: WordNet, OpenCyc and DBpedia. The WordNet and DBpedia concept relatedness evaluation was performed on a number of standard datasets for which the human judgment of relatedness was given. In the case of OpenCyc, we used the same standard datasets as for WordNet and DBpedia, and additionally adapted clustering evaluation techniques to the problem of determining concept relatedness. We evaluated the text annotation framework for WordNet and DBpedia, using data provided by the latest SemEval evaluation workshops.

The concept definition-based measure exhibited high correlation with human judgment for concept pairs where humans assigned a high relatedness score, but had low sensitivity for pairs where humans assigned a low relatedness score. The structure-based measure closely resembled the human judgment of relatedness, having higher sensitivity in the case of concept pairs where humans assigned a low relatedness score compared to the definition-based measure. The hybrid approach

which combines the two types of relatedness measures yielded best results, as the structure-based measure could compensate for the shortcomings of the definition-based measure. Moreover, using the structure-based measure we could reliably recreate predefined concept clusters and generate concept paths which contained less abstract concepts compared to paths generated based on unit weights. The proposed text annotation framework based on concept relatedness obtained competitive results on both WordNet and DBpedia evaluations. This is encouraging as our annotation framework does not make use of additional external corpora. Additionally, rather than taking into account specific characteristics of a particular ontology or knowledge base, the proposed approaches generalize across different ontologies or knowledge bases.

8.1 Scientific Contributions

Automatic text annotation is a challenging task and the dedicated semantic evaluation series (SemEval) aim to advance the state-of-the-art by providing a common evaluation platform. As acquiring semantically-annotated data is still expensive, knowledge-based approaches have become more and more popular, especially with the increase in the number, size and quality of the knowledge bases and ontologies. In this thesis we leverage knowledge-based approaches for automatic text annotation and use different knowledge bases to exemplify the proposed methodology. Our main contributions to the Computational Linguistics and Semantic Web research fields can be summarized as follows:

- Proposing novel approaches to determine the relatedness between concepts defined in background knowledge datasets, which exhibit high correlation with the human judgment of relatedness. We obtain best results which improve over state-of-the-art approaches by combining the concept definitions and dataset structure in a hybrid approach in both the cases of WordNet and DBpedia. For the OpenCyc ontology where few concepts have an associated definition, the structure-based measure provides best results which improve over state-of-the-art approaches.

- Defining a modular and generic automatic text annotation framework which relies on the relatedness between concepts. Our text annotation framework exhibits state-of-the-art performance measured in terms of precision and recall on both WordNet and DBpedia evaluations without requiring additional semantically-annotated corpora. The knowledge base structure is useful for the text annotation task and in the case of DBpedia, results can be further improved by taking into account concept definitions. Choosing a larger local context is generally better compared to choosing a smaller one, provided a robust relatedness measure and aggregation function.

- Applying and evaluating the relatedness measures and the text annotation framework in the case of several background knowledge datasets with different characteristics: WordNet, OpenCyc and DBpedia. This enables the extension of the proposed methodology to other datasets with similar properties.

8.2 Future Work

With respect to future work, we envisage different complementary directions. The text annotation framework can be further extended by integrating other relatedness measures and other types of aggregation functions for relatedness scores.

In the evaluation settings we used well-established cross-domain datasets which enabled us to compare the performance of our approaches against the state-of-the-art. As an alternative, we could evaluate the annotation framework using smaller, domain-specific ontologies or knowledge bases. Moreover, instead of using one background knowledge dataset as input we could take advantage of the interlinks between different Linked Datasets and use a combination of datasets. As the Linked Open Data project develops, the number and quality of the available interlinks should also improve.

Even though we focus on annotating English text, our approach is language independent and can be used to annotate text in other languages, provided there exists an ontology or knowledge base for that language. Future work could test our framework on multilingual knowledge bases such as BabelNet, DBpedia or WikiData.

Another future work direction would be to use our framework in a real-world application. As a first step in this direction (Rusu, Hodson, & Kimball, 2014) we extract events in news articles and obtain a more general representation for the events by linking them to concepts defined in knowledge bases.

Appendix A

Algorithm Implementation

The implementation of the algorithms proposed in this thesis (see Algorithm 3.1, Algorithm 3.2, Algorithm 3.3 and Algorithm 5.1) are available on GitHub [1] at `https://github.com/deliarusu/text-annotation.git`.

The algorithms have been implemented in Python and require the following packages:

- *NLTK* (Natural Language Toolkit), a set of libraries for natural language processing;
- *numpy* a package for scientific computing;
- *NetworkX* a package for complex graph creation and manipulation.

The code is organized in four main packages.

1. The *knowledgebase* package contains modules for representing the knowledge base as a NetworkX graph of concepts and relations between concepts, as well as a module for representing concept definitions as a Bag of Words (BOW).
2. The *text* package is useful for text pre-processing.
3. The *relatedness* package modules contain implementations of the definition-based (see Algorithm 3.1) and structure-based (see Algorithm 3.2, Algorithm 3.3) algorithms proposed in this thesis.
4. The *annotation* package contains modules with the implementation of the text annotation algorithm (see Algorithm 5.1).

The relatedness and text annotation algorithms can be applied to other knowledge bases not described in this thesis by extending the *knowledgebase* package with modules for these knowledge bases.

The *README* file contains more details regarding parameter configuration and software usage.

[1] GitHub `https://github.com` is a code sharing platform.

References

Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., & Pas, M. (2009). A study on similarity and relatedness using distributional and WordNet-based approaches. In *Proceedings of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT)* (pp. 19–27). ACL. Boulder, Colorado, USA.

Agirre, E., De Lacalle, O. L., Fellbaum, C., Hsieh, S.-K., Tesconi, M., Monachini, M., ... Segers, R. (2010). Semeval-2010 task 17: All-words word sense disambiguation on a specific domain. In *Proceedings of the 5th International Workshop on Semantic Evaluation* (pp. 75–80). ACL. Uppsala, Sweden.

Agirre, E. & Rigau, G. (1996). Word sense disambiguation using conceptual density. In *Proceedings of the 16th Conference on Computational Linguistics (COLING)* (Vol. 1, pp. 16–22). Copenhagen, Denmark.

Agirre, E. & Soroa, A. (2007). Semeval-2007 task 02: Evaluating word sense induction and discrimination systems. In *Proceedings of the 4th International Workshop on Semantic Evaluations* (pp. 7–12). ACL. Prague, Czech Republic.

Agirre, E. & Soroa, A. (2009). Personalizing PageRank for word sense disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL)* (pp. 33–41). ACL. Athens, Greece.

Albert, R. & Barabasi, A.-L. (2002). Statistical mechanics of complex networks. *Reviews of Modern Physics*, *74*(1), 47.

Andreopoulos, B., Alexopoulou, D., & Schröder, M. (2008). Word sense disambiguation in biomedical ontologies with term co-occurrence analysis and document clustering. *International Journal of Data Mining and Bioinformatics*, *2*(3), 193–215.

Antoniou, G. & Van Harmelen, F. (2004). *A semantic web primer*. MIT press.

Banerjee, S. & Pedersen, T. (2002). An adapted Lesk algorithm for word sense disambiguation using WordNet. In *Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)* (pp. 136–145). Mexico City, Mexico: Springer-Verlag.

Banerjee, S. & Pedersen, T. (2003). Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI)* (pp. 805–810). Acapulco, Mexico: Morgan Kaufmann Publishers Inc.

Berners-Lee, T. (2006). Linked Data - Design Issues. Retrieved May 2, 2014, from `http://www.w3.org/DesignIssues/LinkedData.html`

Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The semantic web. *Scientific American*, *284*(5), 28–37.

Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with Python*. O'Reilly Media, Inc.

Bizer, C., Heath, T., & Berners-Lee, T. (2009). Linked data-the story so far. *International Journal on Semantic Web and Information Systems*, *5*(3), 1–22.

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

Borg, I. & Groenen, P. J. F. (2005). *Modern Multidimensional Scaling: Theory and Applications (Springer Series in Statistics)*. Springer-Verlag.

Boyd-Graber, J. L., Blei, D. M., & Zhu, X. (2007). A topic model for word sense disambiguation. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)* (pp. 1024–1033). Prague, Czech Republic.

Brin, S. & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, *30*(1), 107–117.

Brody, S. & Lapata, M. (2009). Bayesian word sense induction. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL)* (pp. 103–111). ACL. Athens, Greece.

Bunescu, R. C. & Pasca, M. (2006). Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL)* (Vol. 6, pp. 9–16). Trento, Italy.

Burton-Jones, A., Storey, V. C., Sugumaran, V., & Ahluwalia, P. (2005). A semiotic metrics suite for assessing the quality of ontologies. *Data Knowledge Engineering*, *55*(1), 84–102.

Chan, Y. S., Ng, H. T., & Zhong, Z. (2007). NUS-PT: Exploiting parallel texts for word sense disambiguation in the English all-words tasks. In *Proceedings of the 4th International Workshop on Semantic Evaluations* (pp. 253–256). ACL. Prague, Czech Republic.

Chiarcos, C., Hellmann, S., & Nordhoff, S. (2012). Linking linguistic resources: Examples from the open linguistics working group. In *Linked data in linguistics* (pp. 201–216). Springer-Verlag.

Ciaramita, M. & Altun, Y. (2006). Broad-coverage sense disambiguation and information extraction with a supersense sequence tagger. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 594–602). ACL. Sydney, Australia.

Cilibrasi, R. L. & Vitanyi, P. M. B. (2007). The Google similarity distance. *IEEE Transactions on Knowledge and Data Engineering*, *19*(3), 370–383.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, *20*(1), 37–46.

Collins English Dictionary. (2014). Retrieved May 2, 2014, from `http://www.collinsdictionary.com/dictionary/english`

Collins, A. & Loftus, E. (1975). A spreading-activation theory of semantic processing. *Psychological Review*, *82*(6), 407–428.

Cortes, C. & Vapnik, V. (1995). Support vector machine. *Machine Learning*, *20*(3), 273–297.

Cruz, I. F., Fabiani, A., Caimi, F., Stroe, C., & Palmonari, M. (2012). Automatic configuration selection using ontology matching task profiling. In *Proceedings of the 9th Extended Semantic Web Conference (ESWC)* (Vol. 1, pp. 179–194). Heraklion, Greece.

Cucerzan, S. (2007). Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)* (Vol. 7, pp. 708–716). Prague, Czech Republic.

Cyganiak, R., Wood, D., & Lanthaler, M. (2014). RDF 1.1 Concepts and Abstract Syntax. W3C Recommendation 25 February 2014. Retrieved May 2, 2014, from `http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/Overview.html`

Davies, D. L. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *1*(2), 224–227.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematlk*, *1*, 269–271.

Duda, R. O., Hart, P. E. et al. (1973). *Pattern classification and scene analysis*. Wiley New York.

Euzenat, J. & Shvaiko, P. (2007). *Ontology matching*. Springer-Verlag.

Finkel, J. R., Grenager, T., & Manning, C. (2005). Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (ACL)* (pp. 363–370). ACL. Ann Arbor, Michigan, USA.

Francis, W. N., Kučera, H., & Mackie, A. W. (1982). *Frequency analysis of English usage: lexicon and grammar*. Houghton Mifflin.

Gabrilovich, E. & Markovitch, S. (2007). Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conferences on Artificial Intelligence (IJCAI)* (Vol. 7, pp. 1606–1611). Hyderabad, India.

Genesereth, M. R. & Nilsson, N. J. (1987). *Logical Foundations of Artificial Intelligence*. Morgan Kaufmann Publishers Inc.

Golub, G. H. & Van Loan, C. F. (2012). *Matrix computations*. JHU Press.

Gruber, T. R. (1995). Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human-Computer Studies*, *43*(5), 907–928.

Gutierrez Vazquez, Y. (2012). *Analisis semantico multidimensional aplicado a la desambiguacion del lenguaje natural* (Doctoral dissertation).

Gutierrez Vazquez, Y., Fernandez Orquin, A., Montoyo Guijarro, A., Vazquez Perez, S., et al. (2011). Enriching the integration of semantic resources based on WordNet. *Procesamiento del Lenguaje Natural*, *47*, 249–257.

Gutierrez, Y., Castaneda, Y., Gonzalez, A., Estrada, R., Piug, D. D., Abreu, J. I., ... Camara, F. (2013). UMCC_DLSI: Reinforcing a ranking algorithm with sense frequencies and multidimensional semantic resources to solve multilingual word sense disambiguation. In *Proceedings of the 6th International Workshop on Semantic Evaluation*. ACL. Atlanta, Georgia, USA.

Hirst, G. & St-Onge, D. (1998). Lexical chains as representation of context for the detection and correction malapropisms. In C. Fellbaum (Ed.), *WordNet: An Electronic Lexical Database* (pp. 305–332). MIT Press.

Hliaoutakis, A., Varelas, G., Voutsakis, E., Petrakis, E. G., & Milios, E. (2006). Information retrieval by semantic similarity. *International Journal on Semantic Web and Information Systems*, *2*(3), 55–73.

Hoede, C. (1986). Similarity in knowledge graphs. Memorandum nr. 505, Department of Applied Mathematics, University of Twente, Enschede.

Janowicz, K. & Wilkes, M. (2009). SIM-DLA: A novel semantic similarity measure for description logics reducing inter-concept to inter-instance similarity. In *Proceedings of the 6th Annual European Semantic Web Conference (ESWC)* (pp. 353–367). Hersonissos, Crete, Greece: Springer-Verlag.

Jiang, J. J. & Conrath, D. W. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of International Conference Research on Computational Linguistics (ROCLING X)*.

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the Association for Computing Machinery*, *46*(5), 604–632.

Konrath, M., Gottron, T., Staab, S., & Scherp, A. (2012). SchemEX — Efficient construction of a data catalogue by stream-based indexing of linked data. *Web Semantics: Science, Services and Agents on the World Wide Web*, *16*, 52–58.

Kulkarni, A., Khapra, M. M., Sohoney, S., & Bhattacharyya, P. (2010). CFILT: Resource conscious approaches for all-words domain specific WSD. In *Proceedings of the 5th International Workshop on Semantic Evaluation* (pp. 421–426). ACL. Uppsala, Sweden.

Kulkarni, S., Singh, A., Ramakrishnan, G., & Chakrabarti, S. (2009). Collective annotation of Wikipedia entities in web text. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 457–466). ACM. Paris, France.

Landes, S., Leacock, C., & Tengi, R. I. (1998). Building semantic concordances. In C. Fellbaum (Ed.), *WordNet: An Electronic Lexical Database* (pp. 199–216). MIT Press.

Leacock, C. & Chodorow, M. (1998). Combining local context and WordNet similarity for word sense identification. In C. Fellbaum (Ed.), *WordNet: An electronic Lexical Database* (pp. 265–283). MIT Press.

Leacock, C., Miller, G. A., & Chodorow, M. (1998). Using corpus statistics and WordNet relations for sense identification. *Computational Linguistics*, *24*(1), 147–165.

Lee, Y. K. & Ng, H. T. (2002). An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 41–48). ACL.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., ... Bizer, C. (2014). DBpedia - a large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web Journal. In press.*

Lenat, D. B. (1995). CYC: a large-scale investment in knowledge infrastructure. *Communications of the Association for Computing Machinery*, *38*(11), 33–38.

Lesk, M. (1986). Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the 5th Annual International Conference on Systems Documentation (SIGDOC)* (pp. 24–26). ACM. New York City, New York, USA.

Li, Y., Bandar, Z. A., & McLean, D. (2003). An approach for measuring semantic similarity between words using multiple information sources. *IEEE Transactions on Knowledge and Data Engineering*, *15*(4), 871–882.

Lin, D. (1998). An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning (ICML)* (pp. 296–304). Morgan Kaufmann Publishers Inc.

Lin, D. & Pantel, P. (2002). Concept discovery from text. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING)* (pp. 1–7). ACL.

Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval.* Cambridge University Press.

Mazuel, L. & Sabouret, N. (2008). Semantic relatedness measure using object properties in an ontology. In *Proceedings of the 7th International Semantic Web Conference (ISWC)* (pp. 681–694). Karlsruhe, Germany: Springer-Verlag.

Medelyan, O., Witten, I. H., & Milne, D. (2008). Topic indexing with Wikipedia. In *Proceedings of the First AAAI Workshop on Wikipedia and Artificial Intelligence (WIKIAI)* (pp. 19–24). Chicago, Illinois, USA.

Mendes, P. N., Jakob, M., Garcia-Silva, A., & Bizer, C. (2011). Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems (I-Semantics)* (pp. 1–8). ACM. Graz, Austria.

MeSH. (2014). Retrieved May 2, 2014, from `http://www.nlm.nih.gov/mesh/`

Mihalcea, R. & Csomai, A. (2007). Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management (CIKM)* (pp. 233–242). ACM. Lisbon, Portugal.

Mihalcea, R., Csomai, A., & Ciaramita, M. (2007). Unt-yahoo: Supersenselearner: Combining senselearner with supersense and other coarse semantic features. In *Proceedings of the 4th International Workshop on Semantic Evaluations* (pp. 406–409). ACL. Prague, Czech Republic.

Millers, G. A. & Charles, W. G. (1991). Contextual correlates of semantic similarity. *Language and Cognitive Processes*, *6*(1), 1–28.

Milne, D. & Witten, I. H. (2008a). An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. In *Proceedings of the First AAAI Workshop on Wikipedia and Artificial Intelligence (WIKIAI)*. Chicago, Illinois, USA.

Milne, D. & Witten, I. H. (2008b). Learning to link with Wikipedia. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM)* (pp. 509–518). ACM. Napa Valley, California, USA.

Milne, D. & Witten, I. H. (2013). An open-source toolkit for mining Wikipedia. *Artificial Intelligence*, *194*, 222–239.

Moore, J. L., Steinke, F., & Tresp, V. (2011). A novel metric for information retrieval in semantic networks. In R. Garcia-Castro, D. Fensel, & G. Antoniou (Eds.), *The Semantic Web: ESWC 2011 Workshops* (Chap. 3rd Intern, pp. 65–79). Springer-Verlag.

Morris, J. & Hirst, G. (1991). Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics*, *17*(1), 21–48.

Motta, E., Mulholland, P., Peroni, S., Aquin, M., Gomez-Perez, J. M., Mendez, V., & Zablith, F. (2011). A novel approach to visualizing and navigating ontologies. In *Proceedings of the 10th International Semantic Web Conference (ISWC)* (pp. 470–486). Bonn, Germany: Springer-Verlag.

Navigli, R. (2009). Word sense disambiguation. *Association for Computing Machinery Computing Surveys*, *41*(2), 1–69.

Navigli, R., Jurgens, D., & Vannella, D. (2013). Semeval-2013 task 12: Multilingual word sense disambiguation. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval 2013)* (pp. 222–231). ACL. Atlanta, Georgia, USA.

Navigli, R. & Lapata, M. (2010). An experimental study of graph connectivity for unsupervised word sense disambiguation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(4), 678–692.

Navigli, R., Litkowski, K. C., & Hargraves, O. (2007). Semeval-2007 task 07: Coarse-grained English all-words task. In *Proceedings of the 4th International Workshop on Semantic Evaluations* (pp. 30–35). ACL. Prague, Czech Republic.

Navigli, R. & Ponzetto, S. P. (2012a). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, *193*, 217–250.

Navigli, R. & Ponzetto, S. P. (2012b). Joining forces pays off: Multilingual joint word sense disambiguation. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)* (pp. 1399–1410). Jeju Island, Korea.

Navigli, R. & Velardi, P. (2005). Structural semantic interconnections: A knowledge-based approach to word sense disambiguation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(7), 1075–1086.

Ng, H. T. & Lee, H. B. (1996). Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. In *Proceedings of the 34th Annual Meeting on Association for Computational Linguistics (ACL)* (pp. 40–47). ACL.

Noreen, E. W. (1989). *Computer-intensive methods for testing hypotheses: An introduction.* John Wiley & Sons.

Novischi, A., Srikanth, M., & Bennett, A. (2007). Lcc-wsd: System description for English coarse grained all words task at SemEval 2007. In *Proceedings of the 4th International Workshop on Semantic Evaluations* (pp. 223–226). ACL. Prague, Czech Republic.

Open Multilingual WordNet. (2014). Retrieved May 2, 2014, from `http://compling.hss.ntu.edu.sg/omw/`

OpenCyc. (2014). Retrieved May 2, 2014, from `http://www.cyc.com/platform/opencyc`

Patwardhan, S. (2003). *Incorporating dictionary and corpus information into a context vector measure of semantic relatedness* (Master's thesis, University of Minnesota, Duluth).

Patwardhan, S. & Pedersen, T. (2006). Using WordNet-based context vectors to estimate the semantic relatedness of concepts. In *Proceedings of the Workshop on Making Sense of Sense: Bringing Computational Linguistics and Psycholinguistics Together* (pp. 1–8). ACL. Trento, Italy.

Paulheim, H. & Bizer, C. (2013). Type inference on noisy RDF data. In *Proceedings of the 12th International Semantic Web Conference (ISWC)* (pp. 510–525). Sydney, Australia: Springer-Verlag.

Pedersen, T. (2000). A simple approach to building ensembles of Naive Bayesian classifiers for word sense disambiguation. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics (NAACL)* (pp. 63–69). ACL. Seattle, Washington, USA.

Pesquita, C., Faria, D., Falcao, A. O., Lord, P., & Couto, F. M. (2009). Semantic similarity in biomedical ontologies. *PLoS Computational Biology*, *5*(7).

Pirro, G. (2009). A semantic similarity metric combining features and intrinsic information content. *Data Knowledge Engineering*, *68*(11), 1289–1308.

Ponzetto, S. P. & Navigli, R. (2010). Knowledge-rich word sense disambiguation rivaling supervised systems. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)* (pp. 1522–1531). ACL. Uppsala, Sweden.

Ponzetto, S. P. & Strube, M. (2007). Deriving a large scale taxonomy from Wikipedia. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence* (Vol. 7, pp. 1440–1445). Vancouver, Canada.

Quillian, M. R. (1968). Semantic Memory. In M. Minsky (Ed.), *Semantic Information Processing.* MIT Press.

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.

Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI)* (Vol. 1, pp. 448–453). Morgan Kaufmann Publishers Inc.

Rubenstein, H. & Goodenough, J. B. (1965). Contextual correlates of synonymy. *Communications of the Association for Computing Machinery*, *8*(10), 627–633.

Rusu, D., Fortuna, B., & Mladenić, D. (2011). Automatically annotating text with Linked Open Data. In *The 4th Linked Data on the Web Workshop (LDOW)*. Hyderabad, India.

Rusu, D., Fortuna, B., & Mladenić, D. (2014). Measuring concept similarity in ontologies using weighted concept paths. *Applied Ontology*, *9*(1), 65–95.

Rusu, D., Hodson, J., & Kimball, A. (2014). Unsupervised techniques for extracting and clustering complex events in news. In *The Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation*. Baltimore, Maryland, USA.

Rusu, D. & Mladenić, D. (2014). A framework for annotating text with ontological concepts. *Language Resources and Evaluation. Under review.*

Sahami, M. & Heilman, T. D. (2006). A web-based kernel function for measuring the similarity of short text snippets. In *Proceedings of the 15th International Conference on World Wide Web (WWW)* (pp. 377–386). ACM. Edinburgh, Scotland.

Schreiber, G. & Raimond, Y. (2014). RDF 1.1 Primer. W3C Working Group Note 25 February 2014. Retrieved May 2, 2014, from `http://www.w3.org/TR/2014/NOTE-rdf11-primer-20140624/`

Schütze, H. (1998). Automatic word sense discrimination. *Computational Linguistics*, *24*(1), 97–123.

Schwab, D., Goulian, J., Tchechmedjiev, A., & Blanchon, H. (2012). Ant colony algorithm for the unsupervised word sense disambiguation of texts: Comparison and evaluation. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING)* (pp. 2389–2404). Mumbai, India.

Schwartz, H. A. & Gomez, F. (2011). Evaluating semantic metrics on tasks of concept similarity. In *Proceedings of the 24th International Florida Artificial Intelligence Research Society Conference (FLAIRS-24)*.

Seco, N., Veale, T., & Hayes, J. (2004). An intrinsic information content metric for semantic similarity in WordNet. In *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI)* (Vol. 1, pp. 1089–1090). Valencia, Spain: IOS Press.

SemEval. (2012). Semantic evaluation. Retrieved May 2, 2014, from `http://www.cs.york.ac.uk/semeval-2012/`

SemEval. (2013). Semantic evaluation. Retrieved May 2, 2014, from `http://www.cs.york.ac.uk/semeval-2013/`

SemEval. (2014). Semantic evaluation. Retrieved May 2, 2014, from `http://alt.qcri.org/semeval2014/`

Senseval. (2004). Semantic evaluation. Retrieved May 2, 2014, from `http://www.senseval.org/senseval3`

Sinha, R. S. & Mihalcea, R. (2007). Unsupervised graph-based word sense disambiguation using measures of word semantic similarity. In *Proceedings of the ACM/IEEE International Conference on Software Engineering (ICSE)* (Vol. 7, pp. 363–369). Minneapolis, Minnesota, USA.

Soroa, A., Agirre, E., de Lacalle, O. L., Monachini, M., Lo, J., Hsieh, S.-K., ... Vossen, P. (2010). Kyoto: An integrated system for specific domain WSD. In *Proceedings of the 5th International Workshop on Semantic Evaluation* (pp. 417–420). ACL. Uppsala, Sweden.

Strube, M. & Ponzetto, S. P. (2006). WikiRelate! Computing semantic relatedness using Wikipedia. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 6, pp. 1419–1424).

Sussna, M. (1993). Word sense disambiguation for free-text indexing using a massive semantic network. In *Proceedings of the Second International Conference on Information and Knowledge Management (CIKM)* (pp. 67–74). Washington, DC, USA.

The Gene Ontology Consortium. (2000). Gene ontology: Tool for the unification of biology. *Nature Genetics*, *25*(1), 25–29.

Theoharis, Y., Tzitzikas, Y., Kotzinos, D., & Christophides, V. (2008, May). On graph features of Semantic Web schemas. *IEEE Transactions on Knowledge and Data Engineering*, *20*(5), 692–702.

Toutanova, K., Klein, D., Manning, C. D., & Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (NAACL-HLT)* (pp. 173–180). ACL. Edmonton, Canada.

Tran, A., Bowes, C., Brown, D., Chen, P., Choly, M., & Ding, W. (2010). TreeMatch: A fully unsupervised WSD system using dependency knowledge on a specific domain. In *Proceedings of the 5th International Workshop on Semantic Evaluation* (pp. 396–401). ACL. Uppsala, Sweden.

Tratz, S., Sanfilippo, A., Gregory, M., Chappell, A., Posse, C., & Whitney, P. (2007). PNNL: A supervised maximum entropy approach to word sense disambiguation. In *Proceedings of the 4th International Workshop on Semantic Evaluations* (pp. 264–267). ACL. Prague, Czech Republic.

Tsatsaronis, G., Varlamis, I., & Vazirgiannis, M. (2010). Text relatedness based on a word thesaurus. *Journal of Artificial Intelligence Research*, *37*(1), 1–40.

Tversky, A. (1977). Features of similarity. *Psychological Review*, *84*(4), 327–352.

Van Assem, M., Gangemi, A., & Schreiber, G. (2006). RDF/OWL Representation of WordNet. W3C Working Draft 19 June 2006. Retrieved May 2, 2014, from `http://www.w3.org/TR/wordnet-rdf/l`

Veronis, J. (2004). Hyperlex: Lexical cartography for information retrieval. *Computer Speech & Language*, *18*(3), 223–252.

Vrandečić, D. & Krötzsch, M. (2014). Wikidata: a free collaborative knowledge base. *Communications of the Association for Computing Machinery.*

Wikipedia. (2014). Retrieved May 2, 2014, from `http://en.wikipedia.org/wiki/Wikipedia`

Wu, Z. & Palmer, M. (1994). Verb semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting of the Associations for Computational Linguistics (ACL)* (pp. 133–138). ACL. Las Cruces, New Mexico, USA: Morgan Kaufmann Publishers Inc.

Yang, D. & Powers, D. M. W. (2006). Verb similarity on the taxonomy of WordNet. In *Proceedings of the Third International WordNet Conference (GWC)* (pp. 121–128). Jeju Island, Korea.

Yeh, A. (2000). More accurate tests for the statistical significance of result differences. In *Proceedings of the 18th Conference on Computational Linguistics (COLING)* (pp. 947–953). Saarbrücken, Germany: Morgan Kaufmann Publishers Inc.

Bibliography

Publications Related to the Thesis

Journal Articles

Rusu, D., Fortuna, B., & Mladenić, D. (2014). Measuring concept similarity in ontologies using weighted concept paths. *Applied Ontology*, *9*(1), 65–95.

Rusu, D. & Mladenić, D. (2014). A framework for annotating text with ontological concepts. *Language Resources and Evaluation. Under review.*

Štajner, T., Rusu, D., Dali, L., Fortuna, B., Mladenić, D., & Grobelnik, M. (2010). A service oriented framework for natural language text enrichment. *Informatica (Ljubljana)*, *34*(3), 307–313.

Conference and Workshop Papers

Mladenić, D., Grobelnik, M., Fortuna, B., & Rusu, D. (2012). Text stream processing. In *Proceedings of the 2nd International Conference on Web Intelligence, Mining and Semantics (WIMS)* (p. 5). ACM. Craiova, Romania.

Rusu, D., Fortuna, B., & Mladenić, D. (2009). Improved semantic graphs with word sense disambiguation. In *International Semantic Web Conference (ISWC) Posters&Demos*. Washington, DC, USA.

Rusu, D., Fortuna, B., & Mladenić, D. (2011). Automatically annotating text with Linked Open Data. In *The 4th Linked Data on the Web Workshop (LDOW)*. Hyderabad, India.

Rusu, D., Hodson, J., & Kimball, A. (2014). Unsupervised techniques for extracting and clustering complex events in news. In *The Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation*. Baltimore, Maryland, USA.

Rusu, D., Štajner, T., Dali, L., Fortuna, B., & Mladenić, D. (2010). Demo: Enriching text with RDF/OWL encoded senses. In *International Semantic Web Conference (ISWC) Posters&Demos*. Shanghai, China.

Štajner, T., Rusu, D., Dali, L., Fortuna, B., Mladenić, D., & Grobelnik, M. (2009). A service oriented framework for natural language text enrichment. In *Proceedings of the 12th International Multiconference Information Society (IS)* (pp. 203–206). Ljubljana, Slovenia.

Other Publications

Journal Articles

Rusu, D., Fortuna, B., Grobelnik, M., & Mladenić, D. (2009a). Semantic graphs derived from triplets with application in document summarization. *Informatica (Ljubljana)*, *33*(3).

Trampuš, M., Fuart, F., Pighin, D., Štajner, T., Rusu, D., Stopar, L., ... Grobelnik, M. (2014). DiversiNews: Surfacing diversity in online news. *AI Magazine. Under review.*

Conference and Workshop Papers

Bizau, A., Rusu, D., & Mladenić, D. (2011). Expressing opinion diversity. In *The First International Workshop on Knowledge Diversity on the Web (DiversiWeb)*. Hyderabad, India.

Dali, L., Rusu, D., Fortuna, B., Mladenić, D., & Grobelnik, M. (2009). Question answering based on semantic graphs. In *The Workshop on Semantic Search (SemSearch)*. Madrid, Spain.

Dali, L., Rusu, D., Fortuna, B., Mladenić, D., & Grobelnik, M. (2010). AnswerArt: Contextualized question answering. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD): Part III* (pp. 579–582). Springer-Verlag. Barcelona, Spain.

Dali, L., Rusu, D., & Mladenić, D. (2009). Enhanced web page content visualization with Firefox. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD): Part II* (pp. 718–721). Springer-Verlag. Bled, Slovenia.

Leif Keppmann, F., Flöck, F., Adam, A., Simperl, E., Rusu, D., Holz, G., & Metyger, A. (2012). A knowledge diversity dashboard for Wikipedia. In *ACM Web Science*. ACM. Evaston, IL, USA.

Rusu, D., Dali, L., Fortuna, B., Grobelnik, M., & Mladenić, D. (2007). Triplet extraction from sentences. In *Proceedings of the 11th International Multiconference Information Society (IS)* (pp. 8–12). Ljubljana, Slovenia.

Rusu, D., Fortuna, B., Grobelnik, M., & Mladenić, D. (2008). Semantic graphs derived from triplets with application in document summarization. In *Proceedings of the 10th International Multiconference Information Society (IS)* (pp. 198–201). Ljubljana, Slovenia.

Rusu, D., Fortuna, B., Mladenić, D., Grobelnik, M., & Sipoš, R. (2009b). Document visualization based on semantic graphs. In *Proceedings of the 13th International Conference Information Visualization* (pp. 292–297). IEEE. Barcelona, Spain.

Rusu, D., Fortuna, B., Mladenić, D., Grobelnik, M., & Sipoš, R. (2009c). Visual analysis of documents with semantic graphs. In *Proceedings of the ACM SIGKDD Workshop on Visual Analytics and Knowledge Discovery: Integrating Automated Analysis with Interactive Exploration* (pp. 66–73). ACM. Paris, France.

Trampuš, M., Fuart, F., Berčič, J., Rusu, D., Stopar, L., & Štajner, T. (2013). (i)DiversiNews - A stream-based online service for diversified news. In *Proceedings of the 16th International Multiconference Information Society (IS)* (pp. 184–187). Ljubljana, Slovenia.

Biography

Delia Sorina Rusu was born on October 6, 1984 in Cluj-Napoca, Romania.

After graduating from the Technical University of Cluj-Napoca with an Engineering Degree (Diploma) in Computer Science in 2008, she enrolled in the New Media and E-science doctoral study program at the Jožef Stefan International Postgraduate School in Ljubljana, Slovenia. Her research area is at the intersection between Text Mining, Computational Linguistics and Semantic Web.

During her doctoral studies she was working on different information extraction, summarization, sentiment analysis and word sense disambiguation applications. She contributed to the implementation of a service oriented framework for natural language text enrichment (Enrycher) and a contextualized question answering system (AnswerArt). The main projects funded by the European Union where she was involved were RENDER (Reflecting Knowledge Diversity) and XLike (Cross-Lingual Knowledge Extraction).

Delia was an intern with Google Inc. (Zürich, 2012) where she was working on hierarchical topic models and with Bloomberg L.P. (New York, 2013) where she was working on event extraction from financial news.