FP7-ICT-2011.4.2
Contract no.: 288342
www.xlike.org

Deliverable D3.2.1

Early ontological word-sense-disambiguation prototype

Editor:	Achim Rettinger, KIT
Author(s):	Achim Rettinger, KIT; Lei Zhang, KIT; Delia Rusu, JSI; Blaž Fortuna, JSI; Dunja Mladenić, JSI
Deliverable Nature:	Prototype (P)
Dissemination Level: (Confidentiality)[1]	Public (PU)
Contractual Delivery Date:	M15
Actual Delivery Date:	2.4.2013
Suggested Readers:	All partners using the XLike Toolkit
Version:	1.0
Keywords:	word-sense-disambiguation, semantic annotation, DBpedia, OpenCyc

[1] Please indicate the dissemination level using one of the following codes:
• **PU** = Public • **PP** = Restricted to other programme participants (including the Commission Services) • **RE** = Restricted to a group specified by the consortium (including the Commission Services) • **CO** = Confidential, only for members of the consortium (including the Commission Services) • **Restreint UE** = Classified with the classification level "Restreint UE" according to Commission Decision 2001/844 and amendments • **Confidentiel UE** = Classified with the mention of the classification level "Confidentiel UE" according to Commission Decision 2001/844 and amendments • **Secret UE** = Classified with the mention of the classification level "Secret UE" according to Commission Decision 2001/844 and amendments

Disclaimer

This document contains material, which is the copyright of certain XLike consortium parties, and may not be reproduced or copied without permission.

All XLike consortium parties have agreed to full publication of this document.

The commercial use of any information contained in this document may require a license from the proprietor of that information.

Neither the XLike consortium as a whole, nor a certain party of the XLike consortium warrant that the information contained in this document is capable of use, or that use of the information is free from risk, and accept no liability for loss or damage suffered by any person using this information.

Full Project Title:	XLike – Cross-lingual Knowledge Extraction
Short Project Title:	XLike
Number and Title of Work package:	WP3 – Cross-lingual Semantic Annotation
Document Title:	D3.2.1 Early ontological word-sense-disambiguation prototype
Editor (Name, Affiliation)	Achim Rettinger, KIT
Work package Leader (Name, affiliation)	Achim Rettinger, KIT
Estimation of PM spent on the deliverable:	7 PM

Copyright notice

Executive Summary

The main goal of the XLike project is to extract knowledge from multi-lingual text documents by annotating statements in sentences of a document with a cross-lingual knowledge base. This deliverable will provide the first version of ontology based word-sense-disambiguation with support for knowledge resources handled by early annotation tools from T3.1. The purpose of the early ontological word-sense-disambiguation prototype described here, is to investigate the performance of the ontology based word-sense-disambiguation based on the shallow linguistic processing tools in D2.1.1 with knowledge bases, such as DBpedia and OpenCyc. While this deliverable focuses on providing word-sense-disambiguation of annotations with knowledge resources, the D3.1.2 prototype will employ the result here as part of its functionality and extend the disambiguation of entity mentions by the prediction of relation patterns with knowledge resources.

From now on whenever we use the term “XLike languages” we refer to English, German, Spanish, Chinese, Catalan and Slovenian.

Table of Contents

List of Figures

List of Tables

Abbreviations

NER	Named Entity Recognition
WSD	Word-sense-disambiguation
STA	Slovenian Press Agency
BLP	Bloomberg

1 Introduction

1.1 Motivation

The main goal of the XLike project is to extract formal knowledge from multi-lingual text documents by annotating statements in sentences of a document with a cross-lingual knowledge base. The purpose of the early ontological word-sense-disambiguation prototype described here, is to investigate the performance of shallow multi-lingual text annotation tools with a cross-lingual knowledge bases, such as DBpedia and OpenCyc. While this prototype does only annotate word phrases in the text documents and link them to DBpedia or OpenCyc knowledge resources, the final annotation prototype will extract subject-predicate-object triples (output of D2.2.1 and D2.2.2) and link them to a semantic knowledge representation. Such triples are essential to being able to apply logical constraints specified in a knowledge base or extract semantic graphs, e.g., using event patterns. However, here we do not relate a phrase to other elements e.g. relations and classes, but concentrate on disambiguate entities according to a cross-lingual knowledge base. We consider the word-sense-disambiguation as a subtask of text annotation and the following use cases are related to this task.

1.2 Entity Tracking in Bloomberg Use Case

The Bloomberg.com website maintains a section on market information. As part of the section, each major company has a dedicated page, listing core statistics. The company profile page contains a list of latest news articles, related to the company, pooled from the rest of Bloomberg.com. This works well for major or US companies, for which enough content is produced. However, the list might maintain outdated articles for smaller companies or companies from other parts of the world.

The entity tracking task in the Bloomberg use case is to generate a more up-to-date list of relevant company news, preferably from their home markets for each company. The task can be roughly defined in two steps as (1) detect mentions of the entity (i.e. company), in the multi-lingual news stream and (2) determine which four are most suitable to be displayed on the company news profile page. The first step requires entity extraction from multi-lingual stream, while the second step requires integration and summarization across all languages with relevant articles.

1.3 Topic and Entity Tracking in STA Use Case

STA covers topics related to Slovenia or Slovenian entities (E.g. companies, athletes). As such, tracking relevant news is an important part of editors' daily routine. Technologies developed within XLike project can improve this process by providing tools for detecting relevant articles across languages and media (mainstream, social media).

Formally, topic or entity tracking can be seen as a filter applied to a stream of articles. An article is retained by the filter if it matches the topic, or is related to the entity. Topics can be defined as a standard classification task, with articles on the input and set of matching topics on the output. Entities can be detected using named-entity extractors and text annotation.

For popular topics or entities, the filter can retain a large amount of articles. The information contained within these articles can be visualized or summarized to help the editors in skimming through the content, to identify relevant events.

1.4 Event Identification in STA Use Case

The goal is to develop methodology to identify the event mentioned in the article and describe it with a set of properties (such as time of the event, involved entities, keywords, etc.). The developed algorithms will be able to assign each article to an event. The identified event will be either new (when this will be the first article describing it) or existing (when we have already seen other articles describing it). Events will be stored in an event registry that will provide querying and editing functionality.

An event is defined as a collection of semantic facts/assertions with the focus on actions. The (subject-predicate-object) assertions can be represented as a semantic graph. There are two main steps to detecting events: 1) extracting semantic facts from text (i.e. semantic graph construction) and 2) automatically deciding what set of facts constitutes an event. In order to extract new event patterns from the documents and identify events from the semantic graphs using event patterns, we need to first annotate the documents with knowledge resources, which relies on the word-sense-disambiguation of the annotations.

2 Techniques for word-sense-disambiguation

2.1 Background Knowledge Base

In this section, we will first introduce two knowledge bases that are involved in this deliverable, namely DBpedia and OpenCyc.

2.1.1 DBpedia

DBpedia is a crowd-sourced community effort to extract structured information from Wikipedia and to make this information available on the Web. DBpedia allows you to make sophisticated queries against Wikipedia, and to link other data sets on the Web to Wikipedia data. We hope this will make it easier for the amazing amount of information in Wikipedia to be used in new and interesting ways, and that it might inspire new mechanisms for navigating, linking, and improving the encyclopedia itself.

The English version of the DBpedia knowledge base currently describes 3.77 million things, out of which 2.35 million are classified in a consistent Ontology, including 764,000 persons, 573,000 places (including 387,000 populated places), 333,000 creative works (including 112,000 music albums, 72,000 films and 18,000 video games), 192,000 organizations (including 45,000 companies and 42,000 educational institutions), 202,000 species and 5,500 diseases.

Localized versions of DBpedia in 111 languages are provided. All these versions together describe 20.8 million things, out of which 10.5 million overlap (are interlinked) with concepts from the English DBpedia. The full DBpedia data set features labels and abstracts for 10.3 million unique things in 111 different languages; 8.0 million links to images and 24.4 million HTML links to external web pages; 27.2 million data links into external RDF data sets, 55.8 million links to Wikipedia categories, and 8.2 million YAGO categories. The dataset consists of 1.89 billion pieces of information (RDF triples) out of which 400 million were extracted from the English edition of Wikipedia, 1.46 billion were extracted from other language editions, and about 27 million are data links into external RDF data sets.

2.1.2 OpenCyc

Cyc is an artificial intelligence project that attempts to assemble a comprehensive ontology and knowledge base of everyday common sense knowledge, with the goal of enabling AI applications to perform human-like reasoning. The project was started in 1984 by Douglas Lenat at MCC and is developed by the Cycorp company. Parts of the project are released as OpenCyc, which provides an API, RDF endpoint, and data dump under an open source license.

The latest version of OpenCyc, 4.0, was released in June 2012. OpenCyc 4.0 includes the entire Cyc ontology containing hundreds of thousands of terms, along with millions of assertions relating the terms to each other; however, these are mainly taxonomic assertions, not the complex rules available in Cyc. The knowledge base contains 239,000 concepts and 2,093,000 facts and can be browsed on the OpenCyc website.

The first version of OpenCyc was released in spring 2002 and contained only 6,000 concepts and 60,000 facts. The knowledge base is released under the Apache License. Cycorp has stated its intention to release OpenCyc under parallel, unrestricted licences to meet the needs of its users. The CycL and SubL interpreter (the program that allows you to browse and edit the database as well as to draw inferences) is released free of charge, but only as a binary, without source code. It is available for Linux and Microsoft Windows. The open source Texai project has released the RDF-compatible content extracted from OpenCyc.

2.2 Word-sense-disambiguation with DBpedia

This approach is based on the Named Entities detected by the NERC tools described in D2.1.1 for all XLike languages. On top of that an approach for finding the corresponding DBpedia resources in the target language is deployed. First the name of the detected entity is used to match against the labels of DBpedia resources in the same language and then the "sameAs" links of the resources are used to link to the DBpedia resources in the target language or resources in other datasets. After that we will filter out the inconsistent candidate resources based on the consistency of the types of the resources and the named entities.

In order to recognize entities and disambiguate their meaning, generating DBpedia annotation in text, we will use 5 steps pipeline: 1) named entity recognition by WP2, 2) candidate resource matching, 3) type-based filtering, 4) context-based ranking and 5) linking to other datasets.

In the first step, we will detect named entities using shallow linguistic processing in WP2. Such entities correspond to contiguous tokens and may have a type (LOCATION, ORGANIZITION and PERSON). In WP2, the entities of these three types can be detected.

In the second step, we match the names of the detected entities with the surface forms of the DBpedia resources. The surface forms of DBpedia resources are extracted from the following properties:

- *rdfs:label* offers "chosen names" for resources
- *dbpedia-owl:wikiPageRedirects* offers alternative spellings, aliases, etc.
- *dbpedia-owl:wikiPageDisambiguates* links a common term to many resources

In the next step, we filter out the inconsistent candidate DBpedia resources based on the consistency of the types of the resources and the named entities based on following mapping between the classes of the resources and the types of the named entities:

- *dbpedia-owl:Place* <=> LOCATION
- *dbpedia-owl:Organisation* <=> ORGANIZATION
- *dbpedia-owl:Person* <=> PERSON

After that we will rank the remaining candidates based on the contexts of the mentions and the resources. The assumption is that the more compatible the context of a mention *m* with the description of an entity *e*, the more likely m refers to this specific entity *e*. We consider the description *e.D* of the entity *e* as *dbpedia-owl:abstract* property and the context *m.C* of the name mention *m* as surrounding sentences. The compatibility usually determined by the term co-occurrences between the context of *m* and the description of *e*. We model it as the cosine similarity based on TFIDF between the name mention context and the entity description.

In the last step, we find DBpedia resources in other languages and the same resources in other datasets through *owl:sameAs* property.

2.3 Measuring Concept Similarity in Ontologies

In this deliverable, we also we address the problem of determining the similarity between concepts defined in a knowledge source such as an ontology. We propose a concept similarity algorithm based on geometric models for representing concepts and relationships, which can be applied to different types of ontologies. The key idea is the concept-weighting scheme, which allows for quantifying the degree of abstractness of concepts. The similarity measure can be successfully applied to different types of ontologies. As such, we study and evaluate two ontologies with different characteristics: WordNet and OpenCyc.

Using the proposed measures, which are based on determining the shortest path between two weighted concepts, we could reliably recreate predefined concept clusters. The paths that we generated using our measures contained less infrastructure concepts compared to unit-weight paths. Additionally, we showed that these measures closely resemble the human judgment of similarity. For the details, please refer to the paper "Measuring Concept Similarity in Ontologies using Weighted Concept Paths"[2] in the Annex A.

One application of concept similarity is that of word sense disambiguation, i.e. identifying the corresponding set of concepts, which match a phrase in a given context.

[2] Note that this paper is under review at the moment.

3 Cross-lingual Word-sense-disambiguation Web Services

This section describes the technical implementation of the techniques introduced in the previous section.

3.1 Word-sense-disambiguation Service for DBpedia

This web service takes the output of linguistic processing in WP2 as input, adds the annotations with DBpedia resources on top of the input.

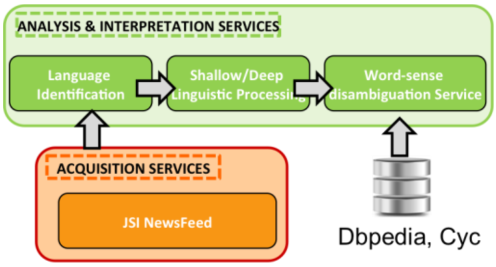
Service Name	**Early Ontological Word-sense-disambiguation Service**
Description	This web service takes the output of multi-linguistic processing in WP2 as input and adds the annotations with knowledge resources, such as DBpedia, OpenCyc etc. by matching the names of the detected entities against the labels of the knowledge resources.
URI	http://km.aifb.kit.edu/services/ner-disambiguation-xx/ (where xx is the language)
Source Code Repository	Will be published to private XLike GitHub repository.
APIs Implemented	Parameters: Output of multi-linguistic processing in WP2
Services Used	 ANALYSIS & INTERPRETATION SERVICES Language Identification Shallow/Deep Linguistic Processing Word-sense disambiguation Service ACQUISITION SERVICES JSI NewsFeed Dbpedia, Cyc The word-sense disambiguation service annotates the entities provided by the multilingual processing analysis with the knowledge resources for the different languages.
Additional Information	None
Notes	None

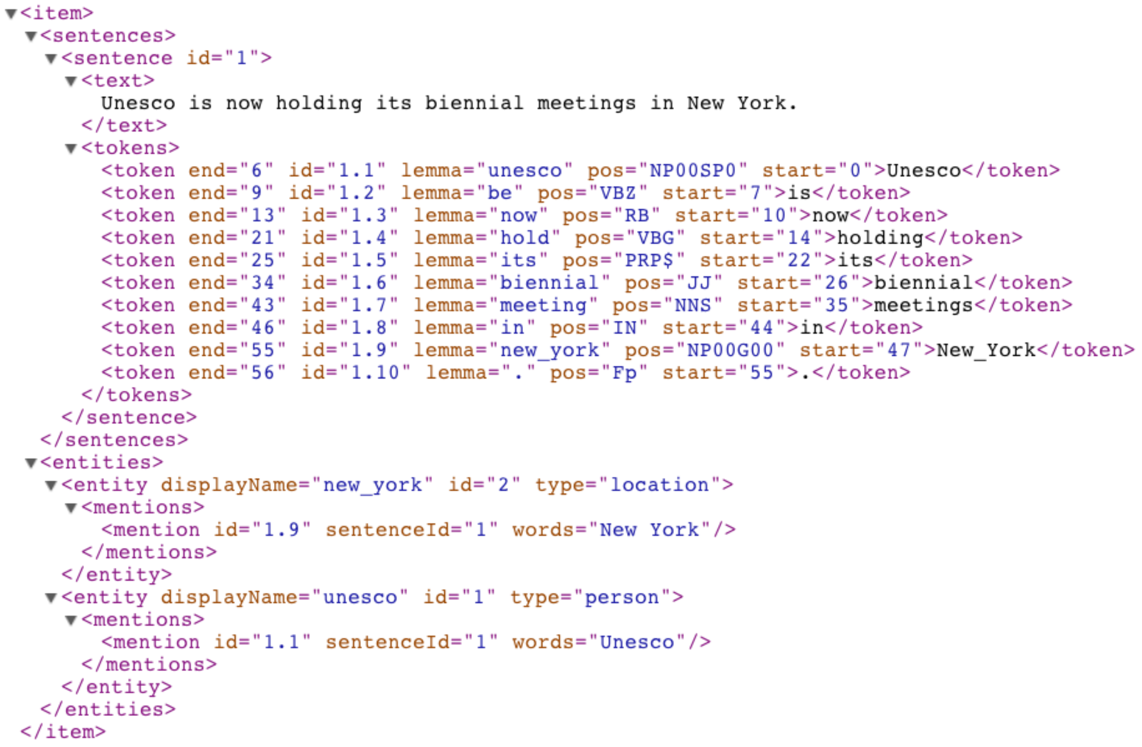
Table 1. Component of Word-sense-disambiguation Service for DBpedia

This web service takes the output of linguistic processing in WP2 as input, adds the Wikipedia annotations by matching the names of the detected entities against the Wikipedia titles.

Language	**URL SandBox**	**Parameters**
English Service	http://km.aifb.kit.edu/services/ner-disambiguation-en/	<item> <sentences> <sentence id=""> <text> </text> <tokens> <token pos=" " end="" lemma=" " id="" start=""> </token>

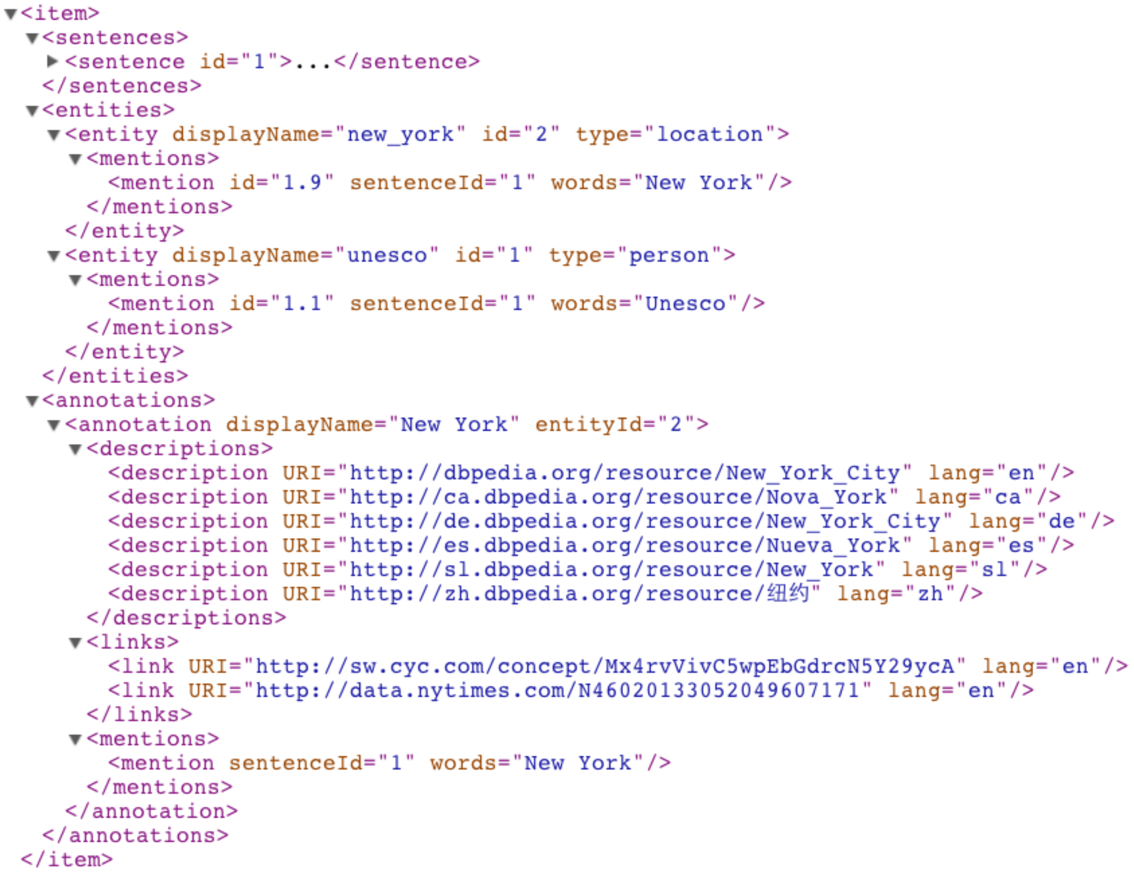
		</tokens> </sentence> </sentences> <entities> <entity type=" " displayName=" " id=""> <mentions> <mention sentenceId="" id="" words=" "></mention> </mentions> </entity> </entities> </item>
Spanish Service	http://km.aifb.kit.edu/services/ner-disambiguation-es/	Same as English Service
Catalan Service	http://km.aifb.kit.edu/services/ner-disambiguation-ca/	Same as English Service
German Service	http://km.aifb.kit.edu/services/ner-disambiguation-de/	Same as English Service
Chinese Service	http://km.aifb.kit.edu/services/ner-disambiguation-zh/	Same as English Service
Slovenian Service	http://km.aifb.kit.edu/services/ner-disambiguation-sl/	Same as English Service

Table 2. Description of Word-sense-disambiguation Service for DBpedia



```
▼<item>
  ▼<sentences>
    ▼<sentence id="1">
      ▼<text>
         Unesco is now holding its biennial meetings in New York.
       </text>
      ▼<tokens>
         <token end="6" id="1.1" lemma="unesco" pos="NP00SP0" start="0">Unesco</token>
         <token end="9" id="1.2" lemma="be" pos="VBZ" start="7">is</token>
         <token end="13" id="1.3" lemma="now" pos="RB" start="10">now</token>
         <token end="21" id="1.4" lemma="hold" pos="VBG" start="14">holding</token>
         <token end="25" id="1.5" lemma="its" pos="PRP$" start="22">its</token>
         <token end="34" id="1.6" lemma="biennial" pos="JJ" start="26">biennial</token>
         <token end="43" id="1.7" lemma="meeting" pos="NNS" start="35">meetings</token>
         <token end="46" id="1.8" lemma="in" pos="IN" start="44">in</token>
         <token end="55" id="1.9" lemma="new_york" pos="NP00G00" start="47">New_York</token>
         <token end="56" id="1.10" lemma="." pos="Fp" start="55">.</token>
       </tokens>
     </sentence>
   </sentences>
  ▼<entities>
    ▼<entity displayName="new_york" id="2" type="location">
      ▼<mentions>
         <mention id="1.9" sentenceId="1" words="New York"/>
       </mentions>
     </entity>
    ▼<entity displayName="unesco" id="1" type="person">
      ▼<mentions>
         <mention id="1.1" sentenceId="1" words="Unesco"/>
       </mentions>
     </entity>
   </entities>
 </item>
```

Figure 1. Example input of the named entity annotation service



```
▼<item>
  ▼<sentences>
    ▶<sentence id="1">...</sentence>
   </sentences>
  ▼<entities>
    ▼<entity displayName="new_york" id="2" type="location">
      ▼<mentions>
         <mention id="1.9" sentenceId="1" words="New York"/>
       </mentions>
     </entity>
    ▼<entity displayName="unesco" id="1" type="person">
      ▼<mentions>
         <mention id="1.1" sentenceId="1" words="Unesco"/>
       </mentions>
     </entity>
   </entities>
  ▼<annotations>
    ▼<annotation displayName="New York" entityId="2">
      ▼<descriptions>
         <description URI="http://dbpedia.org/resource/New_York_City" lang="en"/>
         <description URI="http://ca.dbpedia.org/resource/Nova_York" lang="ca"/>
         <description URI="http://de.dbpedia.org/resource/New_York_City" lang="de"/>
         <description URI="http://es.dbpedia.org/resource/Nueva_York" lang="es"/>
         <description URI="http://sl.dbpedia.org/resource/New_York" lang="sl"/>
         <description URI="http://zh.dbpedia.org/resource/纽约" lang="zh"/>
       </descriptions>
      ▼<links>
         <link URI="http://sw.cyc.com/concept/Mx4rvVivC5wpEbGdrcN5Y29ycA" lang="en"/>
         <link URI="http://data.nytimes.com/N46020133052049607171" lang="en"/>
       </links>
      ▼<mentions>
         <mention sentenceId="1" words="New York"/>
       </mentions>
     </annotation>
   </annotations>
 </item>
```

Figure 2. Example output of the named entity annotation service

4 Evaluation of Word-sense-disambiguation Service

In this section, we present the evaluation results. The experimental setting is same as that in D3.1.1. The only difference is that the evaluation is focused on annotating phrases in non-English documents (source language) and link them to DBpedia resources (target language).

The automatically inserted links to DBpedia resources were manually evaluated by marking the correctness of the links to DBpedia resources either as **yes**, **no** or **0**, where **yes** and **no** were marking the correct or incorrect link respectively and **0** marked the link to the resources corresponding to disambiguation page. In processing of this evaluation results we took the conservative approach and treated **0** answers as **no**, so the calculated precision is representing the completely correct links (i.e, only links marked with **yes**).

First we counted the number of named entities detected in the source documents. Figure 3 shows the difference between the average number of extracted NEs per document by the NER service for English compared to German and Spanish. Compared with evaluation results of Cross-lingual NER Annotation Service in D3.1.1 shown in Figure 4, the number of links is relatively low.

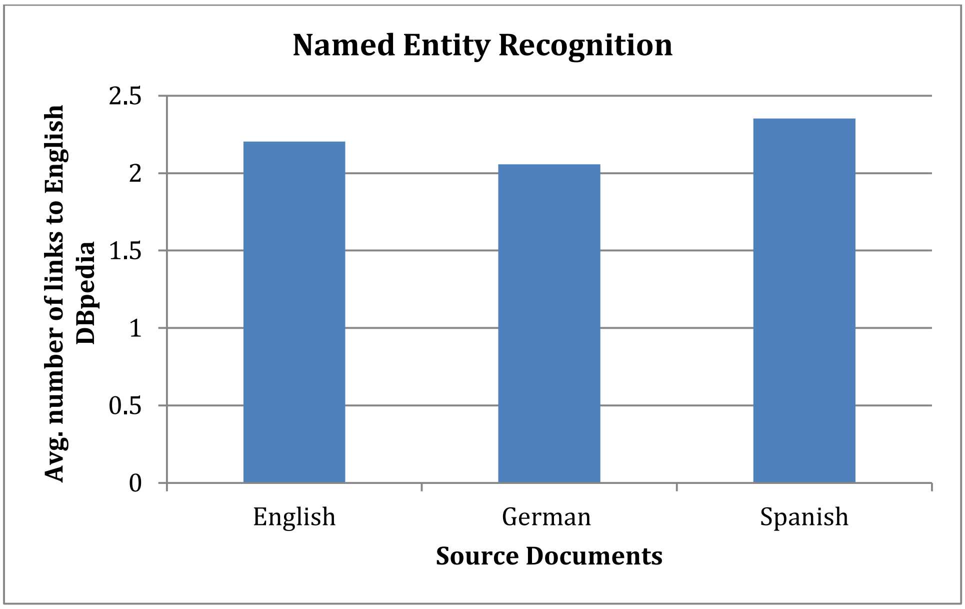


Figure 3. Average number of NE links to DBpedia resources per document

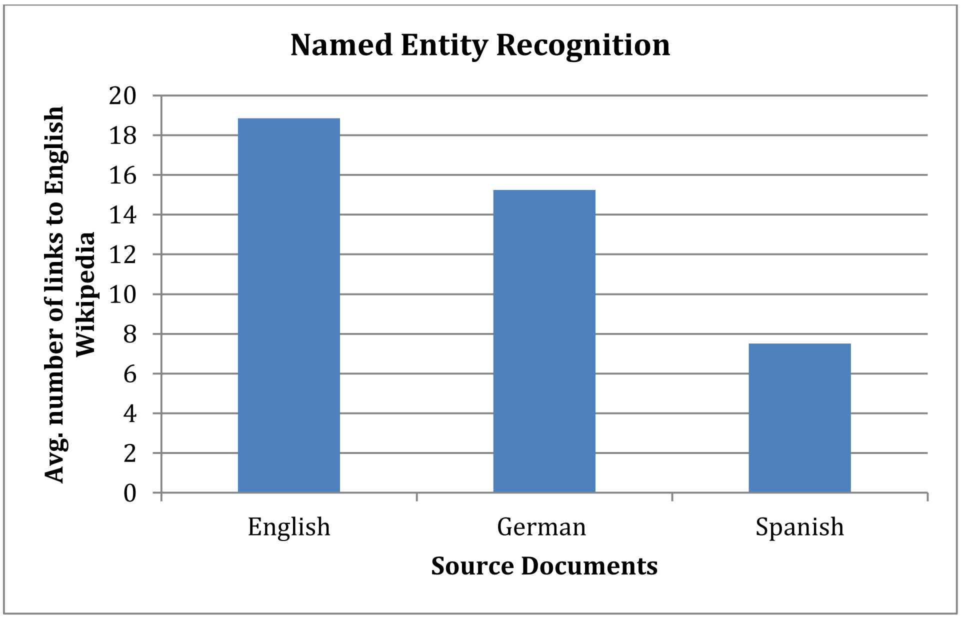


Figure 4. Average number of NE links to Wikipedia pages per document

The precision of links to DBpedia resources is shown in Figure 5. Compared with evaluation results of Cross-lingual NER Annotation Service in D3.1.1 shown in Figure 6, the precision of the links is much higher for all these three languages. Based on the above observation, we believe that the type-based filtering applied in this deliverable heavily decreases the number of NE links to resources in the knowledge base but increases the precision of such links.

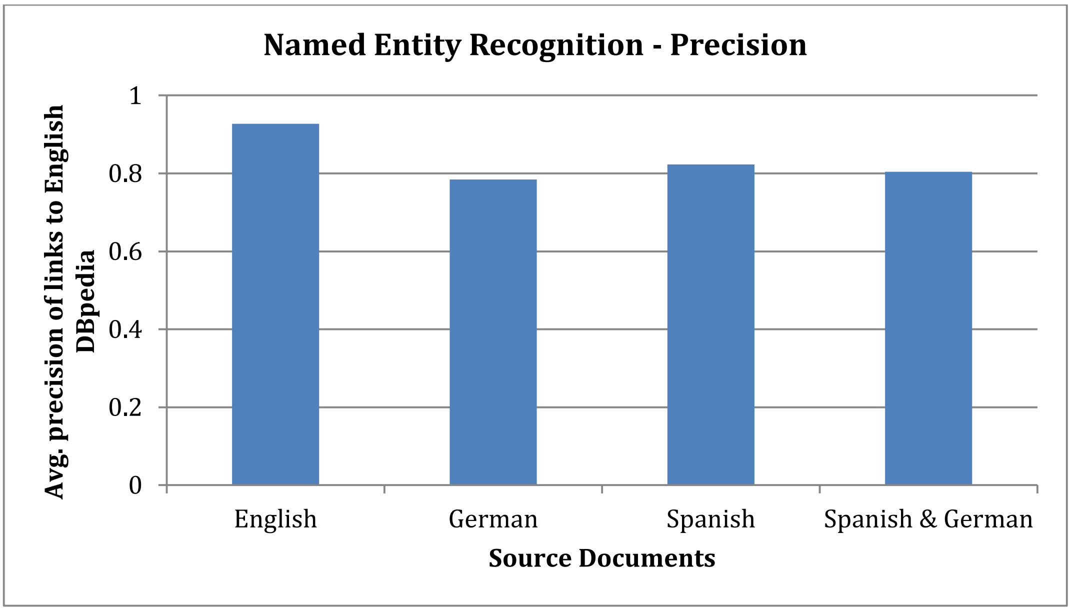


Figure 5. Average precision of NE links to DBpedia resources

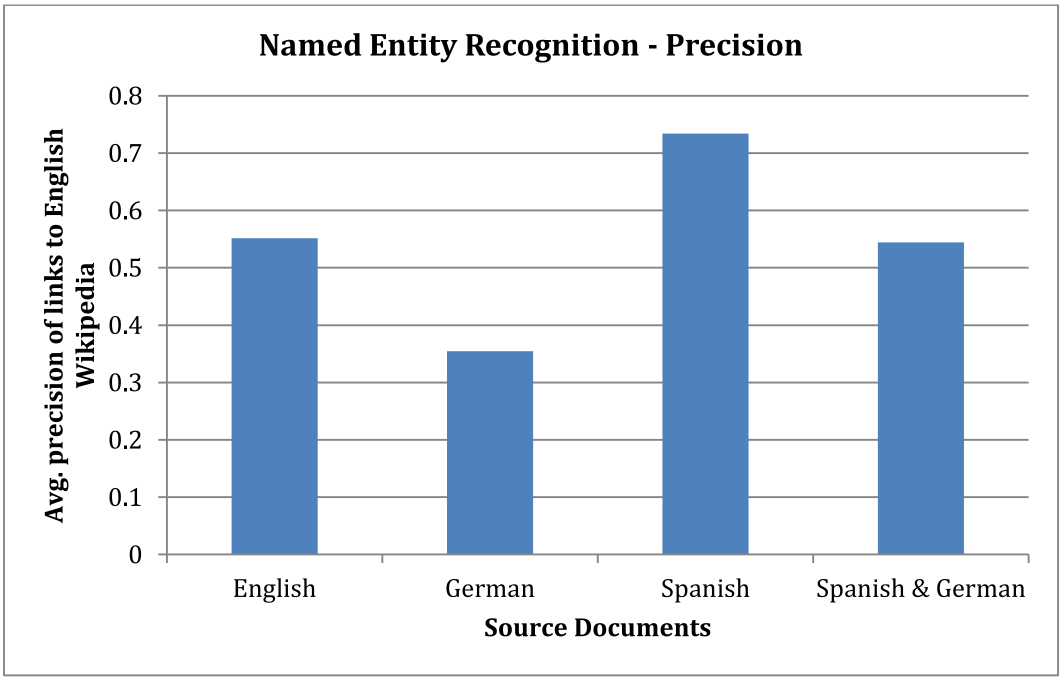


Figure 6. Average precision of NE links to Wikipedia

5 Conclusions

This document presents the Deliverable 3.2.1 Early ontological word-sense-disambiguation prototype. Its structure, functional specification and some details of technical specification are presented. Also, the definition of input, intermediary and output formats are given. The results of the evaluation show that the consistency between the types of detected named entities and the classes of the DBpedia resources can help increase the precision of the links but decrease the number of the links.

References

[D2.1.1] XLike deliverable “D2.1.1 – Shallow linguistic processing prototype”

[D3.1.1] XLike deliverable “D3.1.1 – Early Text Annotation Prototype”

[Lenat1995] Douglas B. Lenat: CYC: A Large-Scale Investment in Knowledge Infrastructure. Commun. ACM 38(11): 32-38 (1995)

[Bizer2009] Christian Bizer, Jens Leh3mann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, Sebastian Hellmann: DBpedia - A crystallization point for the Web of Data. J. Web Sem. 7(3): 154-165 (2009)

Annex A

Measuring Concept Similarity in Ontologies using Weighted Concept Paths

Delia Rusu[3], Blaž Fortuna, Dunja Mladenić
Artificial Intelligence Laboratory, Jožef Stefan Institute
Jožef Stefan International Postgraduate School
Jamova cesta 39, 1000 Ljubljana, Slovenia
{name.surname}@ijs.si

Abstract

Semantic similarity and relatedness between concepts have been extensively studied in different areas ranging from psychology to computational linguistics. In this paper we address the problem of determining the similarity between concepts defined in a knowledge source such as an ontology. We propose a concept similarity algorithm based on geometric models for representing concepts and relationships, which can be applied to different types of ontologies. The key idea is the concept weighting scheme which allows for quantifying the degree of abstractness of concepts. The evaluation settings involving two ontologies validate and highlight the advantages of the proposed approach. Using our measure, which closely resembles the human judgment of similarity, we can reliably recreate predefined concept clusters, and generate more informative concept paths.

Keywords

Semantic similarity and relatedness, weighted concept paths, ontologies.

1. Introduction

Structuring plain-text information is a key prerequisite in coping with information overload. There have been numerous research efforts directed at building structured knowledge sources such as machine readable dictionaries and ontologies. Ontologies formally represent knowledge, usually from a specific domain, as a set of concepts and relationships between concepts. An important task with a long research history and multiple application domains is that of determining the degree of similarity between concepts defined in knowledge sources such as 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 the part-whole relation - *meronymy* or *PART-OF*.

There are numerous applications which take advantage of the similarity or relatedness between ontological 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 et al., 2006), as well as learning based on knowledge sources using different machine learning approaches, e.g. clustering or classification (Milne and Witten, 2012). Another application domain is biomedical and geo-informatics, where concept similarity can be used to compare genes and proteins or geographic features (The Gene Ontology Consortium, 2000).

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, *ontologies* which specify conceptualizations of particular domains or more

[3] Corresponding author. Tel/Fax: +386 1 477 3144, +386 1 477 3851. E-mail: delia.rusu@ijs.si

generic ontologies such as Cyc (Lenat, 1995). The WordNet lexical database (Fellbaum, 1998) and its extensions can be 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:

- *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 et al., 1989; Sussna, 1993; Agirre and Rigau, 1996; Leacock and Chodorow, 1998). Other approaches were based on the distance – i.e. the number of semantic connections - between concepts (Rada et al., 1989; Wu and Palmer, 1994; Leacock and 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 the feature-based model in an information theoretic framework. Semantic similarity was also defined in Description Logics (Janowicz and Wilkes, 2009).

We identify a number of challenges in determining the similarity and relatedness between ontological concepts when utilizing state-of-the-art algorithms (further detailed in Section 2.4). Firstly, methods that provide good results for a given ontology 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 Cyc or DBpedia. Secondly, information content-based measures rely on the probability of occurrence of a concept. These probabilities are usually inferred from frequencies of words in external corpora. Different application domains, however, require different corpora. Moreover, word frequencies and concept frequencies are not equivalent. 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.

This paper addresses the problem of determining the similarity between ontological concepts, using the ontology as a knowledge source. Our aim is to propose a similarity measure which:

- can be successfully applied to different types of ontologies,
- does not require additional corpora aside from the ontology itself,
- can be extended to provide a measure of relatedness between concepts.

The approach we propose relies on the geometric representation of concepts and relationships between them. We distinguish concepts based on their degree of abstractness (Resnik, 1995), i.e. *Entity* would be the most abstract concept in WordNet, as it subsumes all other concepts in the ontology. Next we describe a weighting scheme which can quantify the degree of abstractness of concepts. Our similarity algorithm is based on the notion of *shortest path*, as defined in graph theory. We conduct experiments for both WordNet and OpenCyc, an open source version of Cyc, in order to validate and highlight the advantages of the proposed approach. In the evaluation settings we utilize standard datasets, as well as adapt clustering evaluation techniques to the problem of determining ontological concept similarity.

The paper is structured as follows. In the following subsection 1.1 we start by defining, in more detail, the problem of determining concept similarity in ontologies. Section 2 is dedicated to presenting related work, and comparing between existing measures. We describe two example ontologies having different characteristics in Section 3. In Section 4 we define the concept weighting scheme and describe the concept similarity algorithm. Experimental evaluation is presented in Section 5, while the two final sections of the paper are dedicated to the discussion of results and concluding remarks, respectively.

1.1. The Problem of Determining Concept Similarity in Ontologies

Ontologies specify conceptualizations: objects, concepts, entities from an area of interest and the relationships between them (Genesereth and Nilsson, 1987). Ontologies can differ in structure, way of specifying conceptualizations, and information provided for each concept; this affects the way concept similarity is determined.

Firstly, ontologies are structured in different ways, depending on the purpose for which they are built. Cyc, for example, being a general-purpose ontology, has a number of abstract concepts grouping information. WordNet, on the other hand, is a lexical database containing dictionary-like concepts. If the similarity measure relies on determining the distance between two concepts, an important requirement is that concept distances can be interpreted in a consistent manner (Pirro and 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 0 (Resnik, 1995).

Secondly, the way conceptualizations are specified via ontology classes, instances, object properties, etc. is not consistent across ontologies. For example, the words "friend" and "boy" in the sentence "The two boys are good friends." can be mapped to different WordNet concepts represented via ontology instances of the *NounSynset* class[4]. 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*. 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 thus provide poor results (Rusu et al., 2011).

Table 1 systematizes the characteristics of two ontologies, WordNet and OpenCyc, from the concept similarity perspective.

Table 3 Characteristics of WordNet and OpenCyc which affect concept similarity.

	WordNet	***OpenCyc (subset of Cyc)***
Purpose of the ontology	Lexical database containing dictionary-like concepts	General purpose ontology
How are concepts specified	Via instances of Synset and WordSenses sub-classes	Via classes, instances, object properties (see, for e.g. word "friend")
Number of abstract concepts	Concepts mainly correspond to dictionary terms, the number of abstract concepts being low	Several abstract concepts for grouping information
Concept definitions	Yes	Only for 37% of the classes, instances and object properties
Example sentences containing concepts	Yes	No

2. Related Work

Concept similarity and relatedness have been extensively analyzed within computational linguistics research. Most of the proposed methods have been developed and tested for the WordNet English lexical

[4] RDF/OWL Representation of WordNet [Accessed January 30, 2013]: http://www.w3.org/TR/wordnet-rdf/

database. In what follows, we present some of the most cited approaches, which rely on different characteristics of the ontology.

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 et al., 1982).

2.1. Definition-based Measures

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

Lesk algorithm and its extensions. *Gloss overlap* or the *Lesk algorithm* (Lesk, 1987) is based on computing the word overlap between two or more concept definitions. The algorithm is designed to disambiguate word senses; in this setting, each word has several candidate concepts (equivalent to word senses). The candidate concepts are selected using various techniques, the most straightforward being string matching between the word 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:

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

An extended version of the algorithm, called *extended gloss overlap* (Banerjee and 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 hybrid algorithm considers both the concept definitions, as well as the structure of the ontology. Patwardhan and Pedersen (2006) create second order co-occurrence vectors from concept definitions, called *gloss vectors.* Using relations other than subsumption, the measures proposed in (Banerjee and Pedersen, 2003; Patwardhan and Pedersen, 2006) are considered relatedness measures.

2.2. 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. In what follows, we present the most common measures.

Rada. 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:

$$Distance_{Rada}(c_1, c_2) = minimum\ number\ of\ edges\ separating\ c_1\ and\ c_2.$$

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[5] (MeSH Medical Subject Headings - a hierarchy of medical and biological terms) taxonomies.

Leacock and Chodorow. Another structure-based similarity measure using the distance between two concepts is proposed in (Leacock and Chodorow, 1998). In this case, the shortest path between two concepts is scaled by the depth of the taxonomy, *D.*

[5] MeSH [Accessed January 30, 2013]:http://www.nlm.nih.gov/mesh/

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

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

Wu and Palmer. This measure (Wu and 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 of the two concepts. In a taxonomy such as WordNet, the Least Common Subsumer (LCS) is the closest common ancestor of the two concepts c_1 and c_2.

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

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.

Relatedness measures. Several relatedness measures have been proposed and validated using the WordNet ontology. Hirst and St-Onge describes a relatedness measure centered on the idea of semantically correct paths defined by a set of rules (Hirst and St-Onge, 1998). Each relation type is associated with a direction: ***U****pward*, ***D****ownward* and ***H****orizontal.* 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 and 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 (2006) propose 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*.

2.3. 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 probability of occurrence estimated using noun frequencies from the Brown Corpus of American English (Francis et al., 1982). This corpus provides word frequencies in a collection of texts belonging to genres ranging from news articles to science fiction. The more abstract a concept is, the lower its information content. The *information content (IC)* of a concept *c* is defined as:

$$IC(c) = -\log(p(c))$$

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.

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

Jiang and Conrath. The authors in (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:

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

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

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

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 et al. (2004) introduce the *intrinsic information content*, where the probability of a concept is estimated using the concept hyponyms. This formulation is extended to take advantage of all ontological relations existing between concepts, resulting in the extended information content (Pirro and

Euzenat, 2010). The *extended information content* is defined as a weighted sum of the intrinsic information content and a term that takes into account other relations in the ontology. Together, intrinsic and extended information content are used in a framework inspired from Tversky's feature-based model (Pirro, 2009; Pirro and Euzenat, 2010).

2.4. Comparison between Existing 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 ontologies such as DBpedia or OpenCyc (Rusu et al., 2011). This is due to several reasons. Firstly, concepts in WordNet represent words 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 (Mazuel and 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 rely on the probability of occurrence of a concept in a given corpus. Acquiring these probabilities is a time intensive and expensive process which needs to be repeated whenever the domain changes as different application domains require different corpora. Another problem is that word frequencies and concept frequencies are not equivalent. For example, occurrences of the word "bus" cannot be uniquely mapped onto a single concept, but corresponds 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.

The intrinsic and extended information content-based measures use the ontology itself as a statistical resource. To the best of our knowledge, there has been no research analysing the statistical properties of ontologies, which would motivate these approaches. Moreover, these measures have only been applied in the case of WordNet and MeSH.

3. Example Ontologies: WordNet and OpenCyc

3.1. WordNet

WordNet (Fellbaum, 1998) is a large database of English, containing dictionary-like entries. The main building block of WordNet is the *synset*, an unordered set of cognitive synonyms. The nouns, verbs, adjectives and adverbs are grouped into synsets, where each synset conveys a distinct concept. The synsets are interlinked via a small number of conceptual relations. It also contains a *gloss* (which is a brief definition), example sentences, as well as one or more word senses (see Table 2).

Table 4 A WordNet 3.0 synset for the word "friend".

A noun synset for the word "friend"	supporter, protagonist, champion, admirer, booster, **friend** a person who backs a politician or a team etc. *"all their supporters came out for the game"; "they are friends of the library"*

Word senses	supporter, protagonist, champion, admirer, booster, **friend**
Gloss	a person who backs a politician or a team etc.
Examples	*"all their supporters came out for the game"; "they are friends of the library"*

There are ten relationships defined between synsets, and five between word senses. We provide a more detailed overview of the WordNet 3.0 concepts and relationships between concepts in Table 3. In this work we consider the entire synset as a concept and include both relationships between synsets and word senses.

Table 5 WordNet 3.0 concepts and relationships between concepts.

Concepts		**Relationships between concepts**		
Total Synsets	*117,659*	*Between synsets*	hyponymy, entailment, similarity, member meronymy, substance meronymy, part meronymy, classification, cause, verb grouping, attribute	290,481
Noun Synsets	82,115			
Verb Synsets	13,767			
Adjective Synsets	18,156			
Adverb Synsets	3,621			
Total Word Sense Pairs	*206,941*	*Between word senses*	derivational relatedness, antonymy, see also, participle, pertains to	87,111
Noun Word Sense Pairs	146,312			
Verb Word Sense Pairs	25,047			
Adjective Word Sense Pairs	30,002			
Adverb Word Sense Pairs	5,580			

3.2. OpenCyc

OpenCyc[6] is the open source version of Cyc (Lenat, 1995), a common-sense knowledge base, covering about 40% of the complete Cyc knowledge base. It is also available as a downloadable OWL ontology. In this paper we refer to the 15-08-2010 version of OpenCyc. The OpenCyc OWL ontology includes descriptions of classes, properties (mainly object properties) and instances, each having assigned an RDF[7] resource. 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*.

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 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.

Table 6 OpenCyc OWL 15-08-2010 Version concepts and a subset of relationships between concepts.

Concepts		**Relationships between concepts**	
OWL classes	69,994	*Between an instance and a class*	178,150
Instances	91,287	*Between a class and a superclass*	112,556
		CYC broaderTerm	132,607

[6] OpenCyc [Accessed January 30, 2013]: http://sw.opencyc.org/

[7] Resource Description Framework [Accessed January 30, 2013]: www.w3.org/RDF/

3.3. Illustrative Example: Determining Concept Similarity

We exemplify the task of determining concept similarity in ontologies by referring to two pairs of words: "coast-shore" and "coast-forest". In the experiments of Rubenstein and Goodenough (1965), which we describe in more depth in Section 5.1, these and other word pairs were rated by human assessors, and given a score between 0 and 4. A higher score denotes a higher degree of similarity between the words in a given pair. The word pair "coast-shore" was rated with 3.60, indicating that the words are quite similar, while the pair "coast-forest" was rated with 0.85, indicating that the words are rather dissimilar.

The words "coast", "shore" and "forest" were mapped to the WordNet 3.0 concepts coast *– the shore of a sea or ocean*, shore *– the land along the edge of a body of water*, and forest *– land that is covered with trees and shrubs*. Figure 1 shows a subset of WordNet concepts, as well as some of the relations between them. The concepts *coast* and *shore* are connected via a *hypernymy/hyponymy* relation, and the shortest path between these two concepts comprises only one edge. *Coast* and *forest,* however, are connected via a number of relations, and the shortest path between them comprises four edges. In this case the shortest path determined by counting the edges between the two concepts, coincides to a high degree with the human judgment of similarity.

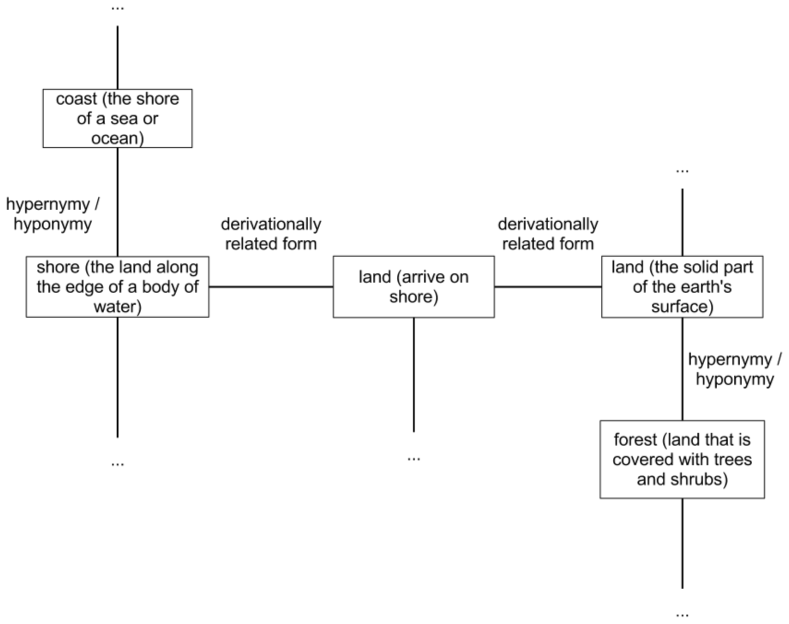

...
coast (the shore of a sea or ocean)
hypernymy / hyponymy
shore (the land along the edge of a body of water)
derivationally related form
land (arrive on shore)
derivationally related form
...
land (the solid part of the earth's surface)
hypernymy / hyponymy
forest (land that is covered with trees and shrubs)
...
...
...

Figure 7 A subset of WordNet 3.0 synsets including word senses and the relationships between them. For each word sense we also show its definition in parenthesis.

Figure 2 shows a subset of OpenCyc concepts, as well as some of the relations between them, for the same example pairs. The word "coast" was manually mapped to the concept *Seacoast*, "shore" was mapped to the concept *Shoreline* and "forest" to the concept *Forest*. The shortest path between *Seacoast* and *Shoreline* is of length one, as there is a direct relation between the two concepts, namely *rdfs:subClassOf*. There are two shortest paths between *Seacoast* and *Forest*: either via the *NaturalThing* concept, or the *NaturalFeatureType* concept. Both shortest paths have a length of two. In this latter case the length of the shortest path does not coincide with the human judgment of similarity.

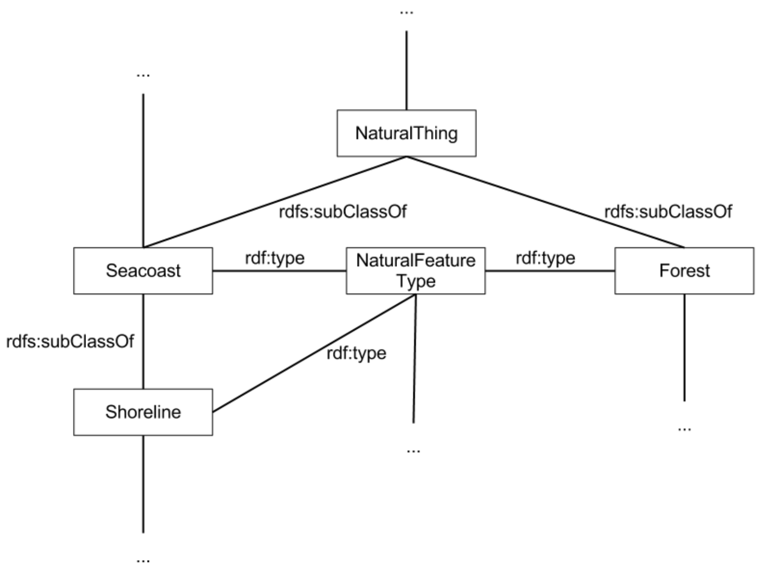


Figure 8 A subset of OpenCyc concepts and the relationships between them.

3.4. Example Application: Word Sense Disambiguation

As previously mentioned, one application of concept similarity is that of word sense disambiguation, i.e. identifying the corresponding set of concepts which match a phrase in a given context.

For example, in the sentence “The two boys are good friends.” the word “boy” can be mapped to three different concepts in WordNet 3.0. For the word “friend” we can identify five different WordNet concepts (see Figure 3). The mapping between words and ontology concepts can be achieved via the natural language identifiers (NLI) of ontological concepts. In Figure 3 the NLIs have been marked in bold, and the matching NLI has been underlined. We can further determine the similarity between each pair of concepts, resulting in 15 such pairwise similarities: (boy_1, $friend_1$), (boy_1, $friend_2$)…(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 similarity value, providing an indication of which pair(s) of concepts is most suitable for disambiguating the example sentence.

1. ***male child***, ***boy*** -a youthful male person 2. ***boy*** -a friendly informal reference to a grown man 3. ***son***, ***boy*** -a male human offspring	1. ***friend*** -a person you know well and regard with affection and trust 2. ***ally***, ***friend*** -an associate who provides cooperation or assistance 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

Figure 9 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.

The focus of this paper is to describe a similarity measure for concepts defined in an ontology. Disambiguating words in text is beyond the scope of this paper.

4. Our Concept Similarity Algorithm based on Weighted Concept Paths

Our approach is based on the geometric model described in cognitive psychology, and inspired from Rada et al.’s work on defining a distance metric on semantic nets (Rada et al., 1989). 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 and 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 ontologies (Resnik, 1995). However, overcoming this drawback is not straight-forward, as different ontologies have very different approaches to defining the concept hierarchy. Take for example WordNet and OpenCyc. WordNet is a dictionary-based taxonomy 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*".

In this work, we propose an extension of the distance metric which is based on assigning weights to ontological concepts and aggregating these weights in an effective manner. We distinguish between two main types of concepts in an ontology:

- **abstract concepts** which have the purpose of structuring information in the ontology – e.g. the concept *Collection* in OpenCyc, and
- **specific concepts** which represent the information in the ontology, and are useful when solving tasks such as automatically annotating text with ontological concepts. Some specific concepts are **information-rich**; these concepts are described using a variety of properties and taxonomic relationships (Motta et al., 2011).

WordNet is organized around specific concepts, some of which are information-rich ones. OpenCyc, on the other hand, also contains a number of abstract concepts for structuring information.

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.

The extension we propose relies on the following two observations.

> **Observation 1 - Weights assigned to concepts.** A weight can be assigned to concepts in such a way as to facilitate distinguishing between abstract and more specific concepts, and also identifying information-rich concepts.
>
> **Observation 2 - Aggregated weights.** The function for aggregating concept weights in order to determine the similarity between two concepts should favor more specific and information-rich concepts.

In what follows, we detail our approach based on the aforementioned two observations. Firstly, we define concept weights such that we can distinguish between the two types of concepts: abstract and more specific. Secondly, we determine the similarity between two concepts by aggregating the concept weights, such that more specific and information-rich concepts are favored.

4.1. Concept Weights

We consider the ontology as a graph *G = (V,E)* where V is the set of all concepts in the ontology, and E represents the relationships between these concepts. The goal is to define a weight associated to each graph node, which would enable distinguishing between node types. 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 et al. (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.

Inspired by the aforementioned previous work, 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.

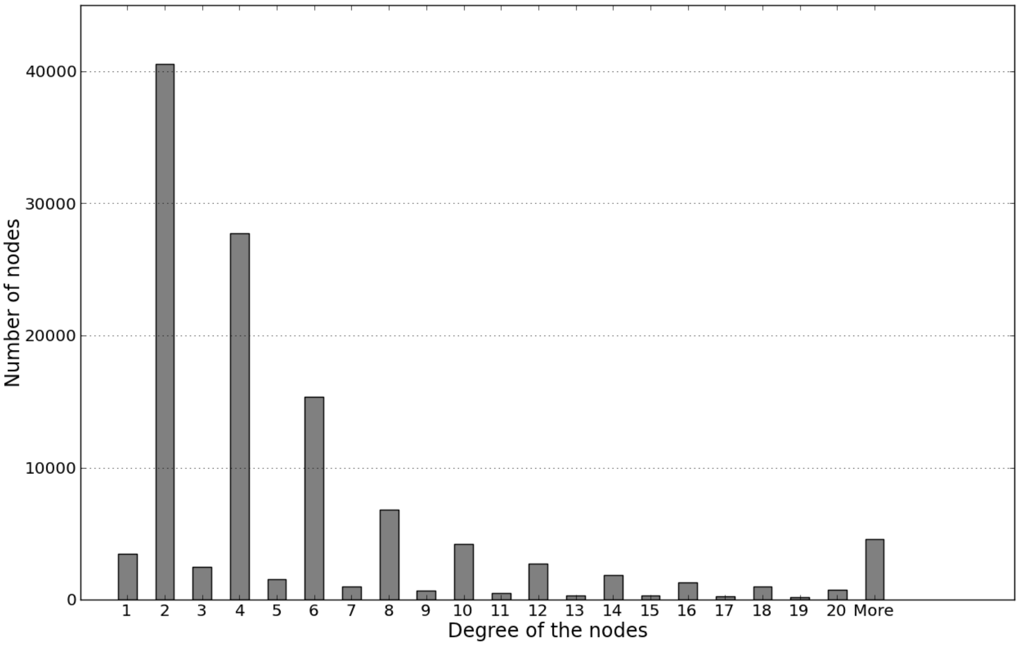


Figure 10 The distribution of node degrees in WordNet 3.0.

Figure 4 shows the distribution of node degrees in WordNet 3.0. WordNet 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.

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. 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. In the case of WordNet, where concepts resemble dictionary entries, the node degree can be interpreted as a weight showing the importance of the node. This allows identifying information-rich nodes.

Figure 5 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 greater than or equal to 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. In the case of OpenCyc, the node degree allows us to differentiate between abstract and more specific concepts.

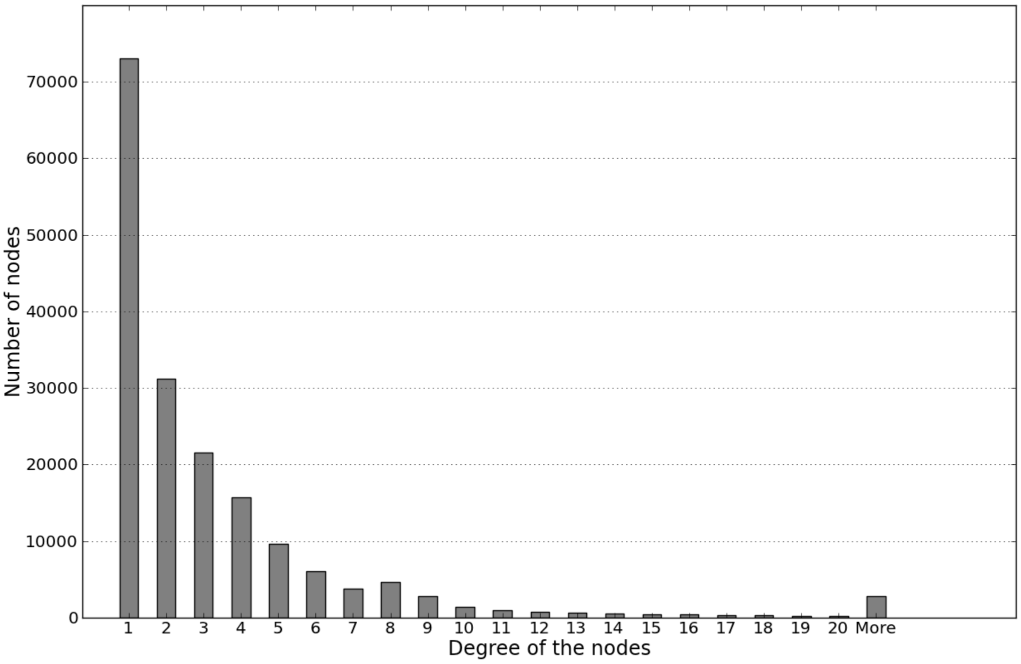


Figure 11 The distribution of node degrees in OpenCyc.

4.2. The Similarity Algorithm

Having decided on the concept 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, one option is to combine the weights of two adjacent nodes into an edge weight. Once the edge weights are computed, we can apply a standard graph algorithm for identifying the shortest path between two nodes. One such algorithm is the Dijkstra algorithm (Dijkstra, 1959). Similar to Rada et al.'s work, the conceptual distance represented by the shortest path between two concepts is a decreasing function of similarity, i.e. the smaller the conceptual distance, the more similar the concepts.

In this paper, we propose a two-step approach for determining the similarity between concepts. As a first step, we combine the weights of adjacent nodes to obtain the edge weight. We want to penalize edges where at least one of the nodes has a high degree. This is due to the fact that high degree nodes are mainly abstract ones, used to structure the ontology content. For the corpora that we used in the evaluation setting (see Section 5), 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 via the Dijkstra algorithm, thereby determining the (weighted) shortest path between two concepts.

The algorithm for computing the conceptual distance between two concepts *c₁* and *c₂*, using weighted concept paths, is described in Figure 6. We start by determining the weight of each node; we propose two such weights in lines 2 and 2', using the logarithm of the degree and the square root respectively. Next, the weight of each edge is found in line 4. Finally, using these edge weights, we apply the shortest path algorithm (e.g. Dijkstra) for each pair of concepts. This yields the conceptual distance, which is a decreasing function of similarity, as shown in line 8.

```
        $Distance_{WeightedConceptPath}(G)$
1           $\boldsymbol{for}\ each\ node\ c\ in\ G$
```

```
            $weight_c = \log(Degree(c))$
            or
            $weight_c = \sqrt{Degree(c)}$
        $\boldsymbol{for}$ $each\ edge\ (c_1, c_2)\ in\ G$
            $weight_{(c_1,c_2)} = Max(weight_{c_1}, weight_{c_2})$
        $\boldsymbol{for}$ $each\ pair\ of\ concepts\ c_1, c_2$
            $\boldsymbol{Distance_{c_1,c_2}} = ShortestPath(c_1, c_2)$

    $Similarity_{WeightedConceptPath}(G)$
        $\boldsymbol{for}$ $each\ pair\ of\ concepts\ c_1, c_2$
                $\boldsymbol{Similarity_{c_1,c_2}} = (-\boldsymbol{1}) \cdot \boldsymbol{Distance_{c_1,c_2}}$
```

Figure 12 The concept similarity algorithm, entitled *Similarity_{WeightedConceptPath}(G),* where G is the graph comprising all concepts and relationships between concepts in the ontology.

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.

For the illustrative example in section 3.3, the concept similarity using weighted concept paths and the log degree weight would yield:

- For **WordNet 3.0**:

$$Similarity_{WeightedConceptPath}(coast, shore) = -Distance_{coast,shore} = -0.30$$

$$\begin{aligned}Distance_{coast,shore} &= ShortestPath(coast, shore) = weight_{(coast,shore)} \\ &= Max(weight_{coast}, weight_{shore}) = Max(\log(27), \log(22)) = 3.30\end{aligned}$$

$$Similarity_{WeightedConceptPath}(coast, forest) = -Distance_{coast,forest} = -14.86$$

$$\begin{aligned}Distance_{coast,forest} &= ShortestPath(coast, forest) \\ &= weight_{(coast,shore)} + weight_{(shore,land_1)} + weight_{(land_1,land_2)} + weight_{(land_2,forest)} \\ &= Max(weight_{coast}, weight_{shore}) + Max\left(weight_{shore}, weight_{land_1}\right) \\ &+ Max\left(weight_{land_1}, weight_{land_2}\right) + Max\left(weight_{land_2}, weight_{forest}\right) \\ &= Max(\log(27), \log(22)) + Max(\log(22), \log(8)) + Max(\log(8), \log(69)) \\ &+ Max(\log(69), \log(20)) = 3.30 + 3.10 + 4.23 + 4.23 = 14.86\end{aligned}$$

- For **OpenCyc**:

$$Similarity_{WeightedConceptPath}(Seacoast, Shoreline) = -Distance_{Seacoast,Shoreline} = -2.48$$

$$\begin{aligned}Distance_{coast,shore} &= ShortestPath(Seacoast, Shoreline) = weight_{(Seacoast,Shoreline)} \\ &= Max(weight_{Seacoast}, weight_{Shoreline}) = Max(\log(5), \log(12)) = 2.48\end{aligned}$$

$$Similarity_{WeightedConceptPath}(Seacoast, Forest) = Distance_{Seacoast,Forest} = -9.98$$

$$\begin{aligned}Distance_{coast,forest} &= ShortestPath(Seacoast, Forest) \\ &= weight_{(Seacoast,NaturalThing)} + weight_{(NaturalThing,Forest)} \\ &= Max\left(weight_{Seacoast}, weight_{NaturalThing}\right) + Max\left(weight_{NaturalThing}, weight_{Forest}\right) \\ &= Max(\log(5), \log(147)) + Max(\log(147), \log(35)) = 4.99 + 4.99 = 9.98\end{aligned}$$

5. Experimental Evaluation

In this section we present experiments on two different ontologies – the first set of experiments is performed using WordNet, while for the second set we use OpenCyc. For both WordNet and OpenCyc we utilize 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.

We compare our proposed methods (described in Section 4.3 and referred to as *WeightedConceptPath Log* and *WeightedConceptPath Sqrt*) to various algorithms from the literature and described in the related work section. We have re-implemented some of those algorithms, in order to apply them to OpenCyc (see Table 5).

In our experiments we represent the concept definitions using a bag-of-words model – i.e. an unordered collection of words. Therefore we can associate with each concept a bag-of-words vector from words belonging to the concept definition and its labels. Then, as suggested in (Banerjee and Pedersen, 2003), the vector is extended with definitions and labels of related concepts. Finally, instead of counting the overlapping words as described in (Lesk, 1987), we compute the cosine similarity between the vectors corresponding to each of the concepts. Cosine similarity is a standard text mining approach to compute the similarity between documents. If c_1 and c_2 are two concepts, and *BOW(c*$_1$*)* and *BOW(c*$_2$*)* are their bag-of-word vectors, then the cosine similarity between the vectors can be defined as:

$$Similarity_{Cos}(BOW(c_1), BOW(c_2)) = \frac{BOW(c_1) \cdot BOW(c_2)}{\|BOW(c_1)\|\|BOW(c_2)\|}$$

This approach is very similar to *gloss vectors* – creating second order co-occurence vectors from concept definitions, as described in (Patwardhan and Pedersen, 2006).

Table 7 A short summary of the re-implemented approaches.

Type of approach	Approach name	Description
Concept definition-based method	*Gloss Overlap*	The concept definitions are represented using bag-of-word vectors, as described further in this section.
Structure-based method	*Wu and Palmer*	The method is based on determining the least common subsumer of the two concepts, as described in Section 3.2
Structure-based methods which determine the shortest path in the ontology	*Shortest Path Unit Weight*	This method determines the distance between two concepts by applying a shortest path algorithm on a unit-weighted graph. $weight(c_1, c_2) = 1$
	Leacock and Chodorow	This method scales the distance between two concepts with the depth of the taxonomy (see Section 3.2).
	Moore et al.	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 c_1 and c_2: $weight(c_1, c_2) = log\big(Degree(c_1)\big) + log\big(Degree(c_2)\big)$

5.1. Experiments using WordNet

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 et al., 2009; Schwartz and 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 assessors. Their judgment was only based on the similarity between the word pairs, all other relationships being disregarded. The MC (Millers and Charles, 1991) dataset 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 to15 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.

In this evaluation setting 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 absolute value of Spearman rank correlations between the systems and human judgment is presented in Table 6.

Table 8 The absolute value of Spearman rank correlations between several systems and the human judgments obtained on three standard datasets (MC, RG and WordSim). The first four systems were proposed in this paper (*WeightedConceptPath Log* and *Sqrt*). The system versions marked with "IS-A" determine the similarity using only IS-A relationships, while the non-marked versions take into account all WordNet 3.0 relationships.

Systems used in the evaluation	**MC** Millers and Charles	**RG** Rubenstein and Goodenough	**WordSim** Finkelstein et al.
WeightedConceptPath Log	0.835	**0.857**	0.667
WeightedConceptPath Log IS-A	0.785	0.811	0.592
WeightedConceptPath Sqrt	0.833	0.827	0.687
WeightedConceptPath Sqrt IS-A	0.804	0.801	0.598
Moore et al.	0.808	0.833	0.650
Moore et al. IS-A	0.792	0.811	0.590
Shortest Path Unit Weight	0.803	0.811	0.601
Shortest Path Unit Weight IS-A	0.775	0.816	0.570
Gloss Overlap	**0.865**	0.811	0.689
Gloss Overlap IS-A	0.858	0.820	**0.694**
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

For the proposed measures, as well as the measures we re-implemented for comparison purposes, we show results when using only *IS-A* relations as well as using all relations available in WordNet 3.0.

To better judge the random error due to small sample size, Table 7 provides the critical values for rejecting the null hypothesis of a system giving a purely random output. The correlation coefficients for the four proposed *WeightedConceptPath* methods exceed the critical values even at the 1% significance level, indicating that these algorithms closely resemble the human judgment of similarity.

Table 9 Sample size and critical values of correlation (two-sided test) for the three datasets (MC, RG, WordSim) used in the evaluation setting.

Dataset	**MC-OpenCyc** Millers and Charles	**RG-OpenCyc** Rubenstein and Goodenough	**WordSim-OpenCyc** Finkelstein et al.
Sample size (number of OpenCyc concept pairs)	28	65	97
Critical value for 5% significance level	+/- 0.374	+/- 0.244	+/- 0.200
Critical value for 1% significance level	+/- 0.478	+/- 0.317	+/- 0.260

5.2. Experiments using OpenCyc

In this sub-section we present two experiments: the first one is based on the standard datasets used in the WordNet experiments, while the second one is performed on a subset of OpenCyc concepts. We annotate the word pairs in the standard datasets with OpenCyc concepts.

5.2.1. Experiments using standard datasets

For our OpenCyc experiments we map the WordNet 3.0 concepts provided in (Schwartz and 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).

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 absolute value of Spearman rank correlations between the aforementioned systems and human judgment is presented in Table 8.

Table 9 provides the critical values for rejecting the null hypothesis of a system giving a purely random output. Similar to the WordNet evaluation results, the correlation coefficients for the four proposed *WeightedConceptPath* methods exceed the critical values even at the 1% significance level, indicating that these algorithms closely resemble the human judgment of similarity.

Table 10 The absolute value of Spearman rank correlations between several systems and the human judgments obtained on three standard datasets (MC, RG and WordSim). The first four systems were proposed in this paper (*WeightedConceptPath Log* and *Sqrt*); the *WeightedConceptPath Log/Sqrt object property* systems determine the similarity between the domain or range of the object property and another concept rather than the object property itself.

Systems used in the evaluation	**MC-OpenCyc** Millers and Charles	**RG-OpenCyc** Rubenstein and Goodenough	**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**
Moore et al.	0.648	0.559	0.356
Shortest Path	0.587	0.304	0.238
Leacock Chodorow	0.587	0.304	0.238
Wu Palmer	0.552	0.390	0.286
Gloss Overlap	0.475	0.341	0.195

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 *ConceptWeightedPath Log* measure is: ***<u>Person</u>*** – ***Teacher*** – ***<u>Prophet</u>*.** The *ConceptWeightedPath Log/Sqrt object property* methods take this observation into account.

Table 11 Sample size and critical values of correlation (two-sided test) for the three datasets (MC, RG, WordSim) used in the evaluation setting.

Dataset	**MC-OpenCyc** Millers and Charles	**RG-OpenCyc** Rubenstein and Goodenough	**WordSim-OpenCyc** Finkelstein et al.
Sample size (number of OpenCyc concept pairs)	20	51	71
Critical value for 5% significance level	+/- 0.444	+/- 0.276	+/- 0.234
Critical value for 1% significance level	+/- 0.561	+/- 0.358	+/- 0.304

5.2.2. 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 centred around information-rich concepts, as described in (Motta et al., 2011).

Our synthetic data consists of 133 randomly chosen OpenCyc concepts belonging to four different categories: 49 names of countries, 35 names of fruits, 21 of computer software and 28 of hardware. Using the methods summarized in Table 5, 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. Other algorithms yield a similarity measure: Leacock and Chodorow, Wu and Palmer, Gloss Overlap. For consistency, the output of the algorithms yielding a similarity measure has been adapted to yield a distance measure by multiplying the results with -1, allowing an easier comparison among algorithms.

In order to validate the results, we propose a clustering approach. 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 would visualize the results, we would expect to identify four different clusters, corresponding to each of the four categories.

For visualizing the results, we use a multidimensional scaling (MDS) approach (Borg and Groenen, 2005). Given the pairwise distances between concepts, MDS assigns each concept a point in the two-dimensional space. Figure 7 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 8 the clustering pattern obtained with the *WeightedConceptPath Log* measure; here we can easily identify the four clusters.

We evaluate the results by means of standard internal clustering evaluation techniques: 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.

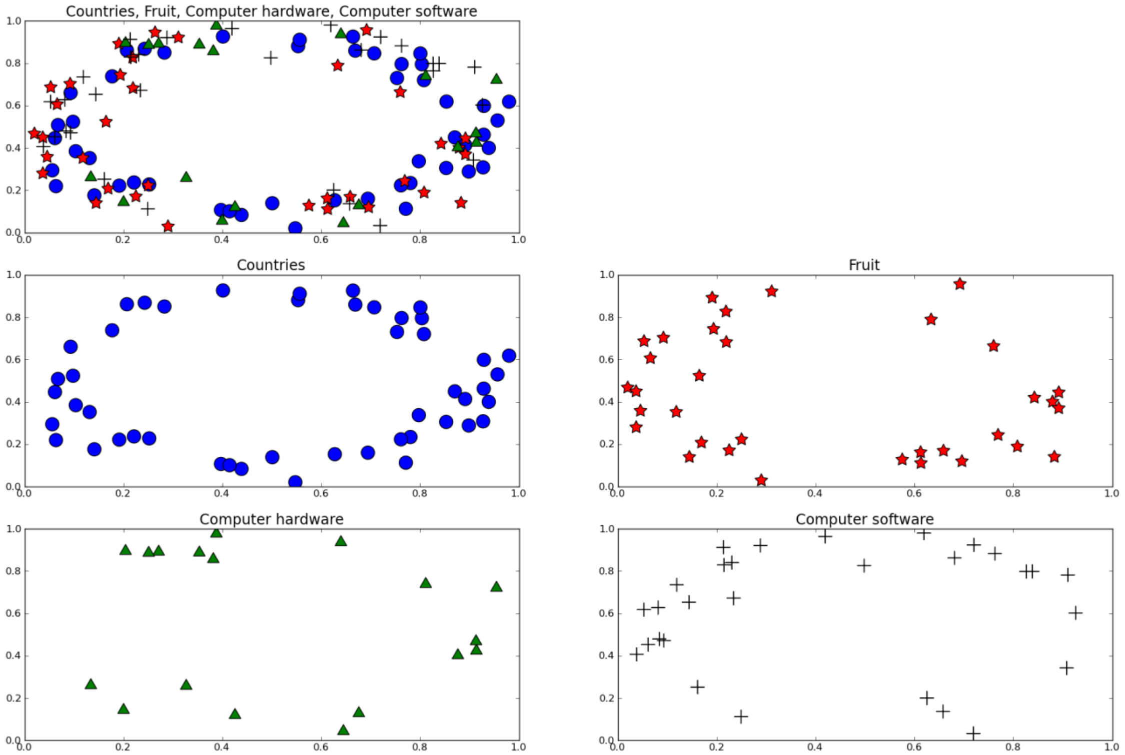


Figure 13 A visualization of concept similarities using the Random measure.

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:

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

and

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

Where N_i is the number of concepts in cluster *i.*

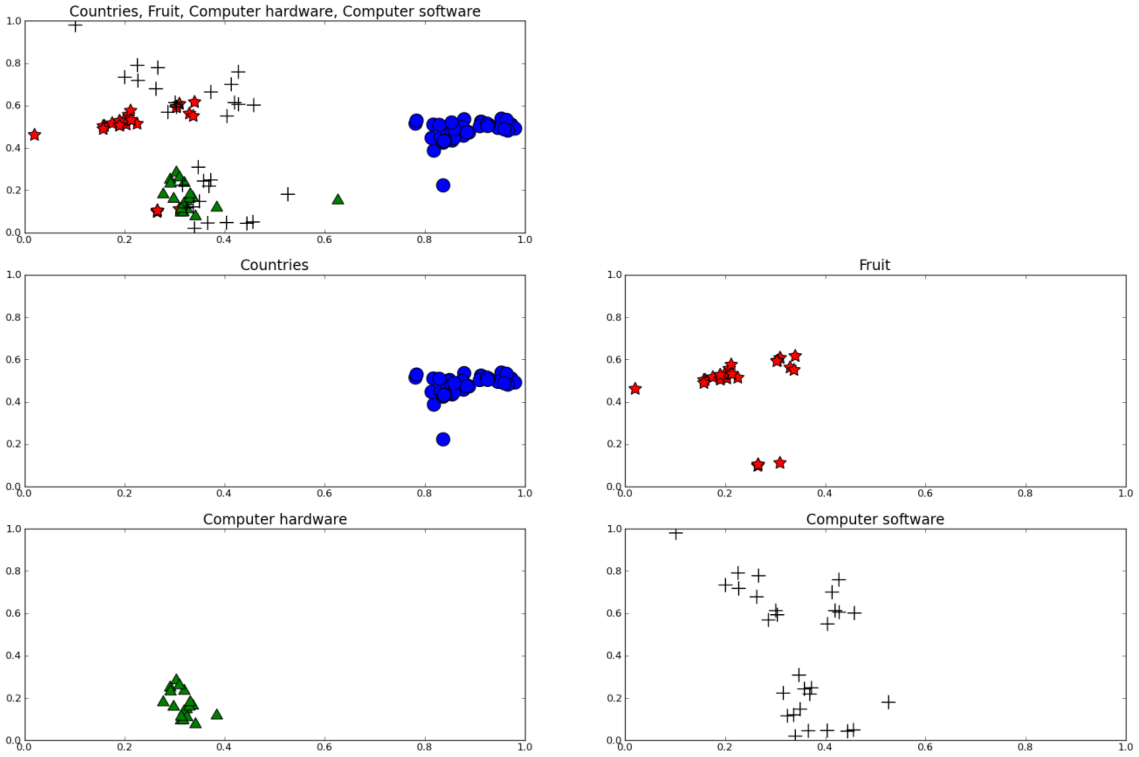


Figure 14 A visualization of concept similarities using the *WeightedConceptPath Log* measure.

Table 10 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*), as well as of the algorithms we compare against.

Table 12 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 133 OpenCyc concepts belonging to four different categories. Our proposed algorithms are *WeightedConceptPath Log* and *Sqrt*, respectively. The best performing algorithms have a low DBI value, low intra-cluster distances and high inter-cluster distances.

	Modified Davies Bouldin Index	**INTRA Cluster Distance**	**INTER Cluster Distance**
WeightedConcept Path Log	**1.363**	13.739	22.513
WeightedConceptPath Sqrt	1.417	29.568	47.745
Moore et al.	1.408	20.153	32.815
Shortest Path	1.653	2.469	3.585
Leacock and Chodorow	1.727	3.469	4.585
Wu and Palmer	1.610	0.123	0.162
Gloss Overlap	1.623	0.582	0.813
Random	1.994	0.497	0.508

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.

6. Discussion

When defining our similarity measure, we highlighted three characteristics of the measure. Firstly, the similarity measure can be successfully applied to different types of ontologies. As such, we chose two ontologies with different characteristics: WordNet and OpenCyc, and presented evaluation results for both ontologies. Secondly, our measure does not require additional corpora aside from the ontology itself. This is an important feature, as we showed that acquiring information from additional corpora is expensive and domain dependent. Finally, we proposed a similarity measure which can be extended to provide a measure of relatedness between concepts.

In the case of WordNet, the experimental evaluation showed improved results when using additional relations aside from the *IS-A* taxonomic links. However, as noted in (Mazuel and Sabouret, 2008), if we use other relations aside from the taxonomic ones, there will be multiple possible paths, some of them not semantically correct. Therefore, we need to explicitly specify constraints for the paths, which we will tackle in future work.

Concept definition-based measures tend to perform well on the Millers and Charles and Rubenstein and Goodenough datasets, when using WordNet as a reference ontology. This is due to the manner in which WordNet was designed - as a lexical database. However, the same results are not reproduceable in the case of OpenCyc, where less than half of the concepts have assigned a definition.

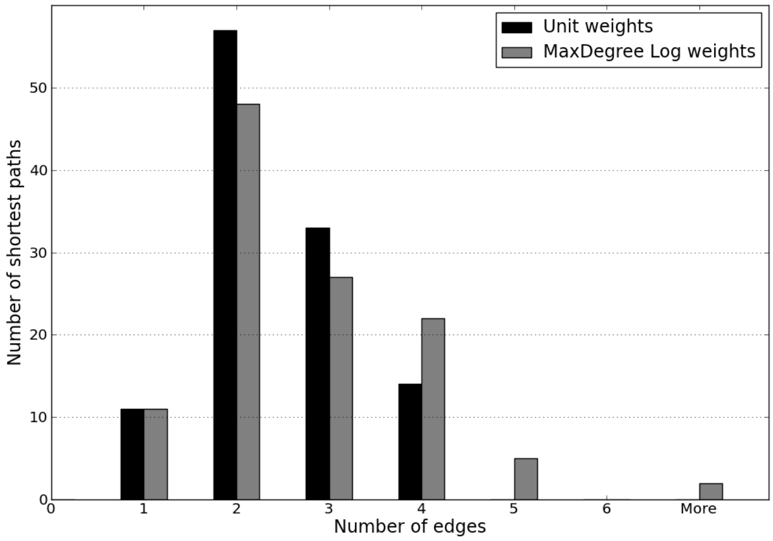


Figure 15 The number of edges in OpenCyc shortest paths, using the unit weights and WeightedConceptPath Log weights.

In general, approaches that use unit weighting in determining the shortest path are outperformed by approaches that employ a weighting scheme based on the ontology characteristics. As a comparison 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. Therefore the unit weight shortest paths are less informative, as they contain more ontology infrastructure nodes with higher node degrees. Figures 9 and 10 graphically depict these observations, using OpenCyc as the underlying ontology.

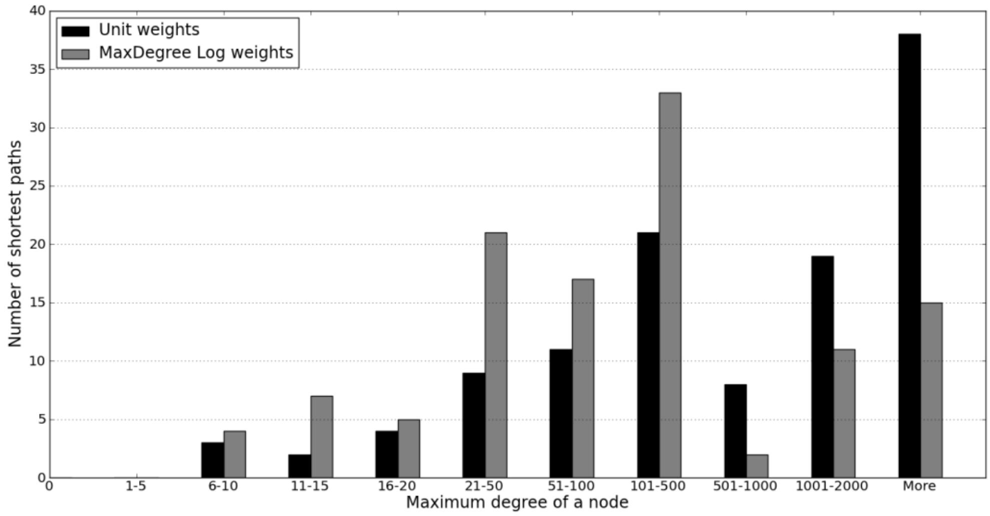


Figure 16 The maximum degree of nodes in OpenCyc shortest paths, using unit weights and WeightedConceptPath Log weights.

As a final discussion point, we focus on the way the OpenCyc ontology was constructed. The ontology construction explains some of the disagreement with human judgment of similarity:

1. **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 similar.
2. **Concepts which are connected via few relationships, and for which humans assign a lower similarity 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 or the word pair “tiger-cat” corresponding to the OpenCyc concepts *Tiger – FelidaeFamily*, which got a 7.35 score.
3. **Concepts that are connected via infrastructure concepts** (with high node degree), e.g. the pair *DividendPaymentObligation – Paying* is connected via *CulturalActivity*, *TemporalStuffType*, the latter having a node degree of 2567.

7. Conclusions and Future Work

In this paper we analyzed the problem of determining the similarity between ontological concepts, using the ontology as a knowledge source. Our analysis presented a number of drawbacks of the similarity measures proposed so far, rooted in not distinguishing between the types of concepts which can appear in an ontology. To overcome this problem, we proposed a concept weighting scheme which defines similarity measures for inconsistent ontologies. In such ontologies the distance between specific and more abstract concepts does not have the same interpretation. We further defined and evaluated two versions of such a similarity measure: *WeightedConceptPath Log and Sqrt*. The evaluation settings highlighted the advantages of these approaches and presented results for two ontologies with different characteristics: WordNet and OpenCyc. The WordNet evaluation was performed on a number of standard datasets for which the human judgment of similarity was given. In the case of OpenCyc, we used the same datasets as for WordNet, and additionally adapted clustering evaluation techniques to the problem of determining concept similarity.

Using the proposed measures which are based on determining the shortest path between two weighted concepts, we could reliably recreate predefined concept clusters. The paths that we generated using our measures contained less infrastructure concepts compared to unit-weight paths. Additionally, we showed that these measures closely resemble the human judgment of similarity.

With respect to future work, we currently envisage three complementary directions. Firstly, the similarity measures can be extended by considering various other paths aside from the shortest paths between concepts. Secondly, we plan to investigate the semantic correctness of the generated paths, when using additional relations in the ontology aside from the taxonomic ones. Thirdly, we plan to test the measures in an application domain, such as word sense disambiguation, and compare them both to purely definition-based measures and hybrid measures.

Acknowledgements

The research leading to these results has received funding from the Slovenian Research Agency and the European Union's Seventh Framework Programme (FP7/2007-2013) RENDER project under grant agreement no.257790.

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 (NAACL). pp. 19–27.

Agirre, E., Rigau, G., 1996. Word sense disambiguation using Conceptual Density, in: Proceedings of the 16th Conference on Computational Linguistics (COLING). Association for Computational Linguistics, Morristown, NJ, USA, pp. 16–22.

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.

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

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

Collins, A., & Loftus, E. (1975). A Spreading-Activation Theory of Semantic Processing. Psychological Review, 82(6), 407–428.

Davies, D.L., 1979. A Cluster Separation Measure. IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-1, 224–227.

Dijkstra, E. W. (1959). A Note on Two Problems in Connexion with Graphs. Numerische Mathematlk, 1, 269–271.

Euzenat, J., Shvaiko, P., 2007. Ontology matching. Springer.

Fellbaum, C. (Ed), 1998. WordNet: An Electronic Lexical Database. MIT Press.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., Ruppin, E., 2010. Placing Search in Context : The Concept Revisited, in: Proceedings of the World Wide Web Conference (WWW). pp. 406–414.

Francis, W.N., Kučera, H., Mackie, A.W., 1982. Frequency Analysis of English Usage: Lexicon and Grammar. Houghton Mifflin.

Genesereth, M.R., Nilsson, N.J., 1987. Logical Foundations of Artificial Intelligence. Morgan Kaufmann.

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

Hliaoutakis, A., Varelas, G., Voutsakis, E., Petrakis, E.G.M., Milios, E., 2006. Information Retrieval by Semantic Similarity. International Journal on Semantic Web and Information Systems 2, 55–73.

Hoede, C. (1986). Similarity in knowledge graphs. Dep. Appl. Math., Twente Univ. of Technology, 7500 AE Enschedc, The Netherlands, Memor. 550.

Janowicz, K., Wilkes, M., 2009. SIM-DL A : 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.

Jiang, J.J., Conrath, D.W., 1997. Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy, in: In Proceedings of International Conference Research on Computational Linguistics (ROCLING X).

Leacock, C., Chodorow, M., 1998. Combining local context and WordNet similarity for word sense identification, in: WordNet: An Electronic Lexical Database. pp. 265–283.

Leacock, C., Millers, G.A., 1998. Using Corpus Statistics and WordNet Relations for Sense Identification. Computational Linguistics 24, 147–165.

Lenat, D.B., 1995. CYC: a large-scale investment in knowledge infrastructure. Communications of the ACM 38, 33–38.

Lesk, M., 1987. 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.

Lin, D., 1998. An Information-Theoretic Definition of Similarity, in: In Proceedings of the 15th International Conference on Machine Learning. pp. 296–304.

Mazuel, L., Sabouret, N., 2008. Semantic relatedness measure using object properties in an ontology, in: Proceedings of the International Semantic Web Conference.

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

Milne, D., Witten, I.H., 2012. An open-source toolkit for mining Wikipedia. Artificial Intelligence. In Press.

Moore, J.L., Steinke, F., Tresp, V., 2011. A Novel Metric for Information Retrieval in Semantic Networks, in: García-Castro, R., Fensel, D., Antoniou, G. (Eds.), The Semantic Web: ESWC 2011 Workshops. Springer Berlin Heidelberg, pp. 65–79.

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 International Semantic Web Conference (ISWC). pp. 470–486.

Navigli, R., 2009. Word sense disambiguation. ACM Computing Surveys 41, 1–69.

Patwardhan, S., Pedersen, T., 2006. Using WordNet-based Context Vectors to Estimate the Semantic Relatedness of Concepts, in: Proceedings of the EACL Workshop on Making Sense of Sense: Bringing Computational Linguistics and Psycholinguistics Together. pp. 1–8.

Pirro, G., 2009. A Semantic Similarity Metric Combining Features and Intrinsic Information Content. Data Knowledge Engineering 68, 1289–1308.

Pirro, G., Euzenat, J., 2010. A Feature and Information Theoretic Framework for Semantic Similarity and Relatedness, in: Proceedings of the International Semantic Web Conference (ISWC). pp. 615–630.

Quillian, M.R., 1968. Semantic Memory, in: Minsky, M. (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, 17–30.

Resnik, P., 1995. Using Information Content to Evaluate Semantic Similarity in a Taxonomy, in: Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI). pp. 448–453.

Rubenstein, H., Goodenough, J.B., 1965. Contextual Correlates of Synonymy. Communications of the ACM 8, 627–633.

Rusu, D., Fortuna, B., Mladenić, D., 2011. Automatically Annotating Text with Linked Open Data, in: Proceedings of the 4th Linked Data on the Web Workshop (LDOW), 20th World Wide Web Conference (WWW).

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: In Proceedings of the European Conference on Artificial Intelligence. pp. 1089–1090.

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.

The Gene Ontology Consortium, 2000. Gene ontology: tool for the unification of biology. Nature Genetics 25, 25–29.

Tversky, A., 1977. Features of Similarity. Psychological Review 84, 327–352.

Wu, Z., Palmer, M., 1994. Verb Semantics and Lexical Selection, in: In Proceedings of the 32nd Annual Meeting of the Associations for Computational Linguistics (ACL). pp. 133–138.

Yang, D., Powers, D.M.W., 2006. Verb Similarity on the Taxonomy of WordNet, in: In Proceedings of the Third International WordNet Conference (GWC). pp. 121–128.