Study of Semantic Relatedness of Words Using Collaboratively Constructed Semantic Resources

Vom Fachbereich Informatik
der Technischen Universität Darmstadt
genehmigte

Dissertation

zur Erlangung des akademischen Grades Dr.-Ing.

vorgelegt von
Dipl.-Inf. Torsten Zesch
geboren in Karl-Marx-Stadt

Tag der Einreichung: 21. Oktober 2009
Tag der Disputation: 1. Dezember 2009

Referenten: Prof. Dr. Iryna Gurevych, Darmstadt
Prof. Dr. Heiner Stuckenschmidt, Mannheim

Darmstadt 2010
D17

Ehrenwörtliche Erklärung [1]

Hiermit erkläre ich, die vorgelegte Arbeit zur Erlangung des akademischen Grades "Dr.-Ing." mit dem Titel "Study of Semantic Relatedness of Words Using Collaboratively Constructed Semantic Resources" selbständig und ausschließlich unter Verwendung der angegebenen Hilfsmittel erstellt zu haben. Ich habe bisher noch keinen Promotionsversuch unternommen.

Darmstadt, den 21. Oktober 2009

Dipl.-Inf. Torsten Zesch

[1] Gemäß §9 Abs. 1 der Promotionsordnung der TU Darmstadt

Wissenschaftlicher Werdegang des Verfassers[2]

10/99–12/05	Studium der Informatik an der Technischen Universität Chemnitz
07/05–12/05	Studienarbeit am Lehrstuhl Datenverarbeitungssysteme Technische Universität Chemnitz "Text Classification Using a Structural Text Model"
06/05–12/05	Diplomarbeit am Lehrstuhl Datenverarbeitungssysteme Technische Universität Chemnitz "Text Classification Based on Conceptual Interpretation"
seit 02/06	Wissenschaftlicher Mitarbeiter am Fachgebiet "Telekooperation" und am Fachgebiet "Ubiquitous Knowledge Processing" an der Technischen Universität Darmstadt

[2]Gemäß §20 Abs. 3 der Promotionsordnung der TU Darmstadt

Abstract

Computing the semantic relatedness between words is a pervasive task in natural language processing with applications e.g. in word sense disambiguation, semantic information retrieval, or information extraction. Semantic relatedness measures typically use linguistic knowledge resources like WordNet whose construction is very expensive and time-consuming. So far, insufficient coverage of these linguistic resources has been a major impediment for using semantic relatedness measures in large-scale natural language processing applications. However, the World Wide Web is currently undergoing a major change as more and more people are actively contributing to new resources available in the so called Web 2.0. Some of these rapidly growing collaboratively constructed resources like Wikipedia and Wiktionary have the potential to be used as a new kind of semantic resource due to their increasing size and significant coverage of past and current developments.

In this thesis, we present a comprehensive study aimed at computing semantic relatedness of word pairs using such collaboratively constructed semantic resources. We analyze the properties of the emerging collaboratively constructed semantic resources Wikipedia and Wiktionary and compare them to classical linguistically constructed semantic resources like WordNet and GermaNet. We show that collaboratively constructed semantic resources significantly differ from linguistically constructed semantic resources, and argue why this constitutes both an asset and an impediment for research in natural language processing. For handling the growing number of available semantic resources, we propose a representational interoperability framework that is used to represent and access all semantic resources in a uniform manner.

We give a detailed overview of the state of the art in computing semantic relatedness and categorize semantic relatedness measures into four types according to their working principles and the properties of the semantic resources they use. We investigate how existing semantic relatedness measures can be adapted to collaboratively constructed semantic resources bridging the observed differences in semantic resources. For that purpose, we perform a graph-theoretic analysis of semantic resources to prove that semantic relatedness measures working on graphs can be correctly adapted. For the first time, we generalize a state-of-the-art vector based semantic relatedness measure to each semantic resource where we can retrieve or construct a textual description for each concept. This generalized semantic relatedness measure turns out to be the most versatile measure being easily applicable to all semantic resources. For the first time, we show (on the example of the German Wikipedia) that the growth of a resource has no or little negative effect on the performance of semantic relatedness measures, but that the coverage steadily increases.

We intrinsically evaluate the adapted semantic relatedness measures on two tasks: (i) comparison with human judgments, and (ii) solving word choice problems. Additionally, we extrinsically evaluate semantic relatedness measures on the task of keyphrase extraction, and propose a new approach to keyphrase extraction based on semantic relatedness measures with the goal to find infrequently used words in a document that are semantically connected to many other words in the document. For the purpose of evaluating keyphrase extraction, we developed a new evaluation strat-

egy based on approximate keyphrase matching that accounts for the shortcomings of exact keyphrase matching. On larger documents, our new approach outperforms all other state-of-the-art unsupervised approaches, and almost reaches the performance of a state-of-the-art supervised approach.

From our comprehensive intrinsic and extrinsic evaluations, we conclude that collaboratively constructed semantic resources provide better coverage than linguistically constructed semantic resources while yielding comparable task performance. Thus, collaboratively constructed semantic resources can indeed be used as a proxy for linguistically constructed semantic resources that might not exist for minor languages.

Zusammenfassung

Die Berechnung der semantischen Verwandtschaft zwischen Wörtern ist von zentraler Bedeutung in der automatischen Sprachverarbeitung und findet Anwendung z.B. in der Lesarten-Disambiguierung, dem semantischen *Information-Retrieval* oder in der Informationsextraktion. Die Maße zur Berechnung der semantischen Verwandtschaft nutzen typischerweise linguistische Ressourcen, wie z.B. WordNet, deren Erstellung sehr zeitaufwändig und teuer ist. Selbst wenn solche linguistischen Ressourcen zur Verfügung stehen, bleibt ihr unzureichender Umfang ein großes Hindernis für die Nutzung von semantischen Verwandtschaftsmaßen in realistischen Anwendungen. Allerdings werden im Zuge der Transformation des World Wide Web ins sogenannte Web 2.0 immer mehr gemeinschaftlich erstellte Ressourcen verfügbar. Beispiele sind Wikipedia und Wiktionary, die sehr schnell wachsen und damit das Potential aufweisen, als neue semantische Ressourcen in der Sprachverarbeitung genutzt zu werden.

In dieser Dissertation untersuchen wir umfassend die Anwendung gemeinschaftlich entwickelter semantischer Ressourcen zur Berechnung der semantischen Verwandtschaft zwischen Wörtern. Dazu analysieren wir die Eigenschaften der gemeinschaftlich entwickelten semantischen Ressourcen Wikipedia und Wiktionary und vergleichen diese mit klassischen, linguistisch motivierten semantischen Ressourcen wie WordNet und GermaNet. Dabei zeigen wir, dass signifikante Unterschiede bestehen, welche einerseits eine Chance zur Erschließung neuen Wissens aus diesen Ressourcen darstellen, es andererseits aber auch notwendig machen, semantische Verwandtschaftsmaße an die gemeinschaftlich erstellten Ressourcen anzupassen. Um die wachsende Anzahl von verfügbaren semantischen Ressourcen effizient handhaben zu können, haben wir ein Interoperabilitäts-Framework entwickelt, in dem alle semantischen Ressourcen einheitlich repräsentiert werden.

Wir geben den Stand der Forschung zu semantischer Verwandtschaft detailliert wieder und kategorisieren existierende Maße in vier Typen, die jeweils unterschiedliche Eigenschaften der semantischen Ressourcen zur Berechnung der semantischen Verwandtschaft nutzen. Wir untersuchen, wie existierende semantische Verwandtschaftsmaße so adaptiert werden können, dass das optimale Zusammenspiel mit gemeinschaftlich erstellten semantischen Ressourcen gewährleistet ist. Zu diesem Zweck führen wir eine graphentheoretische Analyse der semantischen Ressourcen durch und zeigen, dass graphbasierte Maße zur Berechnung semantischen Verwandtschaft korrekt adaptiert werden können. Erstmalig generalisieren wir vektorbasierte Verwandtschaftsmaße auf alle semantischen Ressourcen, welche eine textuelle Beschreibung von Konzepten enthalten oder mit deren Hilfe eine solche Beschreibung konstruiert werden kann. Dieses generalisierte semantische Verwandtschaftsmaß erweist sich in experimentellen Studien bei gleichzeitig hoher Leistung als am vielseitigsten und am einfachsten adaptierbar. Erstmalig zeigen wir (am Beispiel der deutschen Wikipedia), dass das Wachstum einer Ressource keinen oder nur geringen Einfluss auf die Leistung eines semantischen Verwandtschaftsmaß hat, während der Umfang der semantischen Ressource und damit die Einsetzbarkeit in realistischen Anwendungen ständig wächst.

Wir führen eine intrinsische Evaluation der semantischen Verwandtschaftsmaße anhand von zwei etablierten Aufgaben durch: (i) dem Vergleich mit menschlichen

Bewertungen und (ii) der Lösung von Wortauswahlproblemen. Zusätzlich evaluieren wir semantische Verwandtschaftsmaße noch extrinsisch anhand der Eignung zur Extraktion von Schlüsselphrasen. Dazu schlagen wir ein neues Extraktionsverfahren basierend auf semantischen Verwandtschaftsmaßen vor. Durch dieses Verfahren sollen auch Phrasen, welche im Dokument selten vorkommen aber viele semantische Beziehungen zu anderen Wörtern im Dokument besitzen, als Schlüsselphrasen entdeckt werden. Das neue Extraktionsverfahren erweist sich bei längeren Dokumenten allen anderen unüberwachten Verfahren als überlegen und erreicht fast das Leistungsniveau von überwachten Verfahren. Zusätzlich entwickeln wir eine neue Evaluationsstrategie basierend auf einem approximierten Vergleich von extrahierten Schlüsselphrasen mit den vorher annotierten korrekten Schlüsselphrasen. In einer Annotationsstudie zeigen wir, dass diese neue Evaluationsstrategie besser mit menschlichen Bewertungen von Schlüsselphrasen übereinstimmt.

Zusammenfassend lässt unsere umfassende intrinsische und extrinsische Evaluation den Schluss zu, dass gemeinschaftlich entwickelte semantische Ressourcen und linguistische motivierte semantische Ressourcen zu vergleichbaren Ergebnissen führen. Jedoch eignen sich gemeinschaftlich entwickelte semantische Ressourcen durch ihre höhere Abdeckung deutlich besser für realistische Anwendungen. Daher können gemeinschaftlich entwickelte semantische Ressourcen, die für fast alle Sprachen verfügbar sind, als Ersatz für linguistisch motivierte semantische Ressourcen eingesetzt werden, die nur für wenige Sprachen zur Verfügung stehen.

Acknowledgments

I don't know English well enough to properly express how grateful I am to everyone who helped me to finish this thesis. Thus, I am going to write the rest of my acknowledgments in German. If you don't understand German, be sure that your contribution to this thesis is warmly acknowledged.

This work has been supported by the German Research Foundation under grant GU 798/1-2 and GU 798/1-3

Danksagung

Ich danke Prof. Dr. Iryna Gurevych, dass sie einem Informatiker mit rudimentären Kenntnissen der Sprachverarbeitung die Gelegenheit geboten hat, in dem Gebiet zu forschen, welches er spät für sich entdeckt hat. So ziemlich alles, was ich über richtig und falsch in der Wissenschaft weiß, habe ich von ihr gelernt. Ich danke ihr für die beständige und herausfordernde Betreuung meiner wissenschaftlichen Publikationen, die nun in der vorliegenden Arbeit kulminieren. Mein Dank gilt aber auch meinen Kollegen am UKP Lab, die sich immer mit großer Geduld für wissenschaftliche Diskussionen und für endlose Annotationsexperimente haben motivieren lassen.

Hervorzuheben ist auch der Beitrag der wissenschaftlichen Gemeinschaft, die freigiebig Daten geteilt hat, auf Konferenzen ideenreiche Gesprächspartner stellte, und aus der sich das Heer der Gutachter rekrutierte, die durch viele wichtige und richtige Kommentare geholfen haben, meine Publikationen zu verbessern. Besonders möchte ich Anette Hulth, Min-Yen Kan, Alistair Kennedy, Saif Mohammad, Olena Medelyan, David Milne, Giuseppe Pirro, Nuno Seco, Stan Szpakowicz, und Dongqiang Yang dafür danken, dass sie Daten oder Software bereitgestellt haben, die in dieser Arbeit Anwendung fanden.

Es scheint mir kaum möglich, angemessen zu würdigen, auf welch vielfältige Weise meine Eltern zum Entstehen dieser Arbeit beigetragen haben. Ihre bedingungslose Unterstützung meines Weges, der mich weit fort von ihnen geführt hat, war die Grundlage für diese Arbeit.

Mein besonderer Dank gilt Melanie. Ohne dich hätte ich es nicht geschafft.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The lexical cohesion of a text is established by means of lexical-semantic relations between words (Halliday and Hasan, 1976; Morris and Hirst, 1991). For example, *car* is related to *vehicle*, *prize* is related to *Nobel Prize*, and *tree* is related to *leaf*. In these examples, the words are connected by means of classical relations like HYPONYMY (*car* IS-A *vehicle*; *Nobel Prize* IS-A *prize*) or MERONYMY (*tree* HAS-PART *leaf*). However, words can also be connected through non-classical relations (Morris and Hirst, 2004) like functional relationships, co-occurrence, one word being a property of the other words, etc. For example, *car* is related to *drive*, *Albert Einstein* is related to *Nobel Prize*, and *tree* is related to *green*. In the sentence "Albert Einstein did not receive the Nobel Prize for his theory of relativity.", the lexical cohesion of the sentence is almost fully established by non-classical relations between the words *Albert Einstein*, *receive*, *Nobel Prize*, and *theory of relativity*.

Determining the cohesive structure of a text is a pre-requisite of many natural language processing (NLP) applications like word sense disambiguation (Patwardhan et al., 2003), semantic information retrieval (Gurevych et al., 2007), information extraction (Stevenson and Greenwood, 2005), finding real word spelling errors (Budanitsky and Hirst, 2006), and computing lexical chains (Silber and McCoy, 2002; Galley and McKeown, 2003). For most of these applications, it is not necessary to determine the exact type of a relation between two words, but only the strength of the relation, i.e. the **semantic relatedness** between two words.

Humans can easily judge the semantic relatedness between two words. For example, they can easily tell that *car* and *drive* are strongly related, while there is no such strong connection between *car* and *eat*. This human ability is backed by their experience and knowledge, which makes it a hard task for machines. If a machine should solve this task, it also needs some knowledge source. One such knowledge source are large text collections (called *corpora*), where the co-occurrence of two words in the corpus establishes an implicit relation between them. However, in this thesis, we focus on algorithms using knowledge derived from *semantic resources*.[1]

[1]Such resources are often also called *lexical-semantic resources* or *knowledge sources*. For the sake of brevity, we will use the term *semantic resources* instead of *lexical-semantic resources*. However, we are mainly interested in the lexical-semantic knowledge contained in these resources. Thus, we avoid the term *knowledge sources* that is more often used to refer to a resource containing factual knowledge used in artificial intelligence.

Semantic Resources

Semantic resources are knowledge bases containing words (or concepts) and explicitly or implicitly modelled relations between them. A classical example is WordNet (Fellbaum, 1998), an electronic lexical database for the English language that was created by linguists and psycholinguists at Princeton University starting in 1985. WordNet models the English lexicon according to psycholinguistic theory (Miller, 1990; Gross and Miller, 1990; Fellbaum, 1990). The major building block of WordNet are *synsets* (synonym sets), i.e. sets of words that are synonyms, e.g. (*auto*, *automobile*, *car*, *machine*, *motorcar*). Each synset represents a concept and is normally linked with other synsets by semantic relations, e.g. HYPONYMY expressing an IS-A relation between two concepts like in (*car – jeep*). Each synset is accompanied by a GLOSS that gives a short definition of the concept.

However, it is very expensive and time-consuming to create such resources, and they usually cover only a limited number of relations. Thus, we investigate the applicability of other semantic resources that are collaboratively constructed on the Web like Wikipedia and Wiktionary. Wikipedia is a multilingual, Web based, freely available *encyclopedia*, constructed in a collaborative effort of voluntary contributors. Wiktionary is designed as the lexical companion to Wikipedia, being a multilingual, Web based, freely available *dictionary*, *thesaurus* and *phrase book*. An analysis of Wikipedia and Wiktionary reveals that different parts of these resources reflect different aspects of conventional semantic resources, and that Wikipedia and Wiktionary are largely complementary. As these resources are freely available and quickly growing, they constitute a possible substitute for conventional semantic resources like WordNet. In this thesis, we are going to investigate the applicability of collaboratively constructed semantic resources for computing the semantic relatedness between words.

Semantic Relatedness Measures

The algorithms used for determining the strength of a relation using semantic resources are called *semantic relatedness measures*. In order to determine the strength, they use a certain property of a semantic resource. We categorize semantic relatedness measures into four types according to their working principles and the properties of the semantic resources they use:

Path based measures rely on paths in a graph built from a semantic resource. The nodes of this graph represent the words contained in the resource. Two nodes are connected by an edge, if the semantic resource contains a relation between the two words. The length of the shortest path between two words indicates how related they are.

Information content based measures are similar to path based measures, but in addition to the length of a path, information content based measures also take into account how informative a certain word is. For example, the word *Porsche* tells you more about its properties than naming it just *car*. The more specific a word, the higher its information content. The specificity of a word can be measured using its corpus frequency or its position in the semantic graph.

Gloss based measures determine the strength of the relationship between two words by measuring the overlap between glosses. *Glosses* are short definitions of a concept that are usually contained in semantic resources.

Vector based measures construct a vector representation of a word. The semantic relatedness of two words is then determined by computing the cosine similarity between the vector representations.

Many semantic relatedness measures have been defined for a certain type of lexical-semantic information in a specific semantic resource, e.g. the gloss based measure on short definitions from dictionaries, or the vector based measure using a vector space built from Wikipedia articles. However, due to structural or semantic differences between semantic resources, an adaptation process might be necessary if a semantic relatedness measure should be used with other semantic resources. We focus on adapting path and information content based measures to collaboratively constructed semantic resources. For this purpose, we perform a graph-theoretic analysis of semantic resources. We also adapt gloss based measures to all semantic resources focusing on the effects of definition length and quality. Furthermore, we generalize vector based measures operating on Wikipedia to other semantic resources.

Evaluation Framework

We evaluate semantic relatedness measures using two intrinsic and one extrinsic task. The first intrinsic task is comparison with human judgments. The judgments are collected by presenting the judges a list of word pairs. For each word pair, every judge rates the semantic relatedness on a certain scale (e.g. $\{0, 1, 2, 3, 4\}$ where 0 means 'no relatedness' and 4 means 'maximum relatedness'). The single scores are then averaged over all judges to obtain the final human judgment for a word pair. The pair (*car – drive*) might get an average human judgment of 3.7, while another word pair (*car – eat*) might only get 1.1. To evaluate the performance of a semantic relatedness measure, the scores computed by a measure are correlated with the human judgments. The higher the correlation, the better the measure. In previous work, two correlation measures have been widely used: (i) Pearson product-moment correlation coefficient r, and (ii) Spearman rank order correlation coefficient ρ. In Chapter 5, we discuss advantages and disadvantages of both correlation measures in detail, and recommend Spearman rank order correlation as the more suitable measure.

The second intrinsic evaluation task is solving word choice problems. A word choice problem consists of a target word and four candidate words or phrases. The objective is to pick the one that is most closely related to the target. An example problem is given below. There is always only one correct candidate, 'a)' in this case.

beret

a) round cap	b) cap with horizontal peak
c) wedge cap	d) helmet

The semantic relatedness between the target 'beret' and each of the candidates is computed by a semantic relatedness measure, and the candidate with the maximum

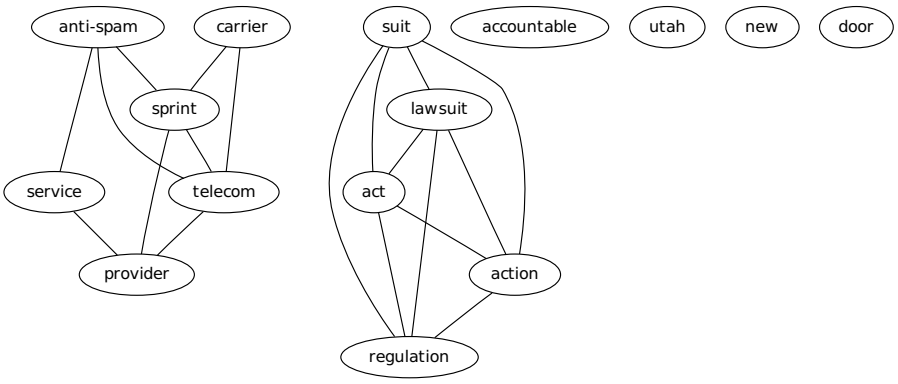


Figure 1.1: Example of a lexical-semantic graph.

semantic relatedness value is chosen. The more word choice problems a semantic relatedness measure is able to solve, the better is its performance.

In addition to the two intrinsic evaluation tasks, semantic relatedness measures should also be extrinsically tested in an application. This will show whether the differences between measures, found in the intrinsic evaluation, have an impact on the performance in a real-life task. We select the task of keyphrase extraction to extrinsically evaluate semantic relatedness measures. Keyphrases are small sets of expressions representing the content of a document. Keyphrase extraction is the task of automatically extracting such keyphrases from a document.

We propose a new approach to keyphrase extraction that is based on measuring the semantic relatedness between terms in a document. Most other approaches use the frequency of a word in a text as a very important clue to decide whether the word is a keyphrase or not. However, due to reading and writing economy, terms might not be repeated in a document (Barker and Cornacchia, 2000). We hypothesize that even if a word occurs only once in a document, it might be of high importance if it is semantically related to many other terms in the same document. To find such important concepts, we construct a *lexical-semantic graph* (LSG). In an LSG, edges represent the strength of the semantic relatedness between two terms. For example, the document

> Anti-spam suit attempts to hold carriers accountable.
> A lawsuit alleges that Sprint has violated Utah's new anti-spam act. The action could open the door to new regulations on telecommunication service providers.

is represented by means of a LSG as shown in Figure 1.1. As the lexical-semantic graph is a fully connected graph that cannot be easily visualized, we only show edges representing strong relationships. We also assume for the sake of the example that the LSG was created using a semantic relatedness measure which perfectly determines the relationships between the words in the document. The resulting LSG would then contain two clusters corresponding to *lawsuit* related words and *telecommunication* related words. The keyphrases for that document will be selected with high probability from both clusters thus covering both topics.

Main Contributions

We now give a brief summary of the main contributions of this thesis:

- We give a comprehensive overview of the properties of the emerging collaboratively constructed resources Wikipedia and Wiktionary. We compare these new semantic resources with linguistically constructed semantic resources like WordNet or GermaNet. We also describe the state of the art in computing semantic relatedness, and categorize existing algorithms into four distinct types of measures, where each type uses different properties of the underlying semantic resources.

- We propose a representational interoperability framework that is used to represent and access all semantic resources in a uniform manner. Algorithms for determining the semantic relatedness between words are then only implemented once using the interoperability framework instead of being adapted to each semantic resource.

- We show how existing semantic relatedness measures can be adapted to collaboratively created semantic resources bridging the observed differences in semantic resources. For that purpose, we perform a graph-theoretic analysis of semantic resources to prove that semantic relatedness measures working on graphs can be correctly adapted.

- For the first time, we generalize the vector based semantic relatedness measure called *ESA* (Gabrilovich and Markovitch, 2007) to each semantic resource where we can retrieve or construct a textual description for each concept. This generalized semantic relatedness measure turns out to be the most versatile measure being easily applicable to all semantic resources.

- For the first time, we analyze (on the example of the German Wikipedia) the influence of the growth of a semantic resource over time on the performance when computing semantic relatedness. We show that the growth has no or little negative effect on the performance of semantic relatedness measures, but that the coverage steadily increases.

- We perform comprehensive experiments with semantic relatedness measures using two evaluation approaches: (i) *comparison with human judgments* and (ii) *solving word choice problems*. We find that collaboratively constructed semantic resources can fully substitute classical linguistically constructed semantic resources.

- We select keyphrase extraction as an extrinsic evaluation task for semantic relatedness measures and propose a new approach to keyphrase extraction based on semantic relatedness measures with the goal to find infrequently used words in a document that are semantically connected to many other words in the document. For the purpose of evaluating keyphrase extraction, we developed a new evaluation strategy for keyphrase extraction based on approximate keyphrase matching that accounts for the shortcomings of exact keyphrase matching. On larger documents, our new approach outperforms all

other state-of-the-art unsupervised approaches and almost reaches the performance of a state-of-the-art supervised approach.

- We developed a set of enabling software required for this thesis that were not publicly available beforehand: (i) the Wikipedia application programming interface *JWPL*, and (ii) a system for semi-automatically creating datasets for the evaluation of semantic relatedness measures called *DEXTRACT*. JWPL is widely used for accessing Wikipedia for research purposes and has also been commercially licensed. Additionally, we augmented the UIMA software component repository DKPro to enable the experiments performed in this thesis. We also implemented (i) a representational interoperability framework for semantic resources called *Lexical-Semantic Resource Interface* , and (ii) a generalized framework for keyphrase extraction.

Publication Record

This thesis builds on a number of publications in peer-reviewed conference and workshop proceedings from major events in natural language processing and artificial intelligence, e.g. ACL, NAACL, AAAI, EMNLP, RANLP, LREC and GLDV/GSCL. The publications (Zesch and Gurevych, 2006), (Zesch and Gurevych, 2007), and (Zesch et al., 2007b) jointly describe the evaluation framework, the adaptation of semantic relatedness measures to Wikipedia, and the graph-theoretic analysis of semantic resources. These papers are summarized in an article in the peer-reviewed Journal of Natural Language Engineering (Zesch and Gurevych, 2010).

In (Zesch et al., 2007a) and (Zesch et al., 2008a), we analyze the properties of Wikipedia and Wiktionary and introduce them as valuable semantic resources for various natural language processing related tasks including computing semantic relatedness. For the first time, we apply Wiktionary to the task of computing semantic relatedness in (Zesch et al., 2008b). In the same paper, we introduce the generalized vector based semantic relatedness measure. In (Zesch and Gurevych, 2009), we use keyphrase extraction as an extrinsic evaluation task and introduce a new evaluation metric based on approximate matching.

We also actively contributed to the following publications in which the work described in this thesis had an impact on other research: Mohammad et al. (2007) investigate a cross-lingual distributional approach to compute semantic relatedness. Müller et al. (2008b), Hartmann et al. (2008), and Culo et al. (2009) describe applications of the semantic relatedness measures from this thesis in semantic information retrieval, context-aware user interfaces, and lexical chaining algorithms. Garoufi et al. (2008a) build on the graph-theoretic analysis from Zesch and Gurevych (2007) and augment it to other semantic resources. Garoufi et al. (2008b) describe the representational interoperability framework that is used to represent and access all semantic resources in a uniform manner.

Thesis Outline

In this thesis, we present a comprehensive study aimed at computing semantic relatedness of word pairs using collaboratively constructed semantic resources. In

Chapter 2, we analyze the properties of the emerging collaboratively created semantic resources Wikipedia and Wiktionary and compare them to classical linguistically created semantic resources like WordNet or GermaNet. We show that collaboratively created semantic resources significantly differ from linguistically created semantic resources, and argue why this constitutes both an asset and an impediment for research in NLP. In Chapter 3, we give an overview of the state of the art in computing semantic relatedness, and in Chapter 4, we investigate how existing semantic relatedness measures can be adapted to collaboratively created semantic resources bridging the observed differences in semantic resources. For that purpose, we perform a graph-theoretic analysis of semantic resources to prove that semantic relatedness measures working on graphs can be correctly adapted. In Chapter 5, we describe the evaluation framework that is used to test the adapted measures on two intrinsic tasks: (i) comparison with human judgments, and (ii) solving word choice problems. In Chapter 6, we present the results of this intrinsic evaluation. In Chapter 7, we extrinsically evaluate semantic relatedness measures on the task of keyphrase extraction. We conclude with a summary in Chapter 8. Appendix A describes the enabling technologies developed in order to conduct our experiments. Appendix B gives an augmented overview of the experimental results.

Chapter 2

Semantic Resources

Many natural language processing (NLP) tasks including computing semantic relatedness require external sources of lexical-semantic knowledge such as dictionaries, thesauri, or wordnets. **Dictionaries** (e.g. the Longman Dictionary of Contemporary English (Procter, 1978)) list all lexical entities in a domain, connect them with their semantic meaning via a defining gloss, and enumerate all senses in case of polysemous entities. Like a dictionary, a **thesaurus** (e.g. Roget's Thesaurus (Berrey and Carruth, 1962)) lists lexical entities, but additionally categorizes them into topical groups by means of semantic relations like SYNONYMY, HYPERNYMY, or MERONYMY. A **semantic wordnet** (e.g. WordNet (Fellbaum, 1998) or GermaNet (Kunze, 2004)) displays features of the aforementioned simpler resources: Like a dictionary, it offers an account of lexical units, their senses and sometimes even short glosses. Additionally, lexical units and senses are organized in a thesaurus structure. Furthermore, Ruiz-Casado et al. (2005) have proposed to add encyclopedic features to wordnets by augmenting WordNet entries with Wikipedia articles. **Encyclopedias** (e.g. Encyclopædia Britannica)[1] offer a detailed description of each lexical entry, but few explicitly modeled relations.

Traditionally, these resources have been built manually by experts in a time consuming and expensive manner. Recently, emerging Web 2.0 technologies have enabled user communities to collaboratively construct new kinds of resources. Wikipedia[2] and Wiktionary[3] are instances of semantic resources that are collaboratively constructed by mainly non-professional volunteers on the web. We call such semantic resources **Collaboratively Constructed Semantic Resources** (CSRs), as opposed to **Linguistically Constructed Semantic Resources** (LSRs) like WordNet or GermaNet.

In this chapter, we first compare LSRs and CSRs on a general level in Section 2.1. We then describe linguistically constructed semantic resources in Section 2.2, and collaboratively constructed semantic resources in Section 2.3. Finally, we introduce a representational interoperability framework for semantic resources in Section 2.4, and conclude with a summary in Section 2.5.

[1] `http://www.britannica.com`
[2] `http://www.wikipedia.org`
[3] `http://www.wiktionary.org`

	LSRs	CSRs
Constructors	Linguists	Mainly non-professional volunteers
Construction approach	Following theoretical model or corpus evidence	Following non-binding guidelines
Construction costs	Significant	None
Data quality	Editorial control	Social control by the community
Available languages	Major languages	Many interconnected languages
Up-to-dateness	Quickly out-dated	Mostly up-to-date
Size	Limited by construction costs	Huge or quickly growing

Table 2.1: Comparison of linguistically and collaboratively constructed semantic resources.

2.1 Comparison of Semantic Resources

Wikipedia and Wiktionary are instances of collaboratively constructed semantic resources (other examples are OpenThesaurus[4] or Yahoo!Answers[5]). The properties of such CSRs differ from LSRs in several ways – Table 2.1 gives an overview. LSRs are typically constructed by linguists following some theoretical model or guided by corpus evidence, while CSRs are constructed by non-professional volunteers that follow non-binding guidelines. The collaborative construction approach results in certain advantages:

- CSRs are released under a license that grants free usage, while LSRs are usually more restrictively distributed due to their very costly construction and maintenance process (except for WordNet that is also freely available);
- CSRs are mostly up-to-date while the release cycles of LSRs cannot reflect recent events or development of a language;
- popular CSRs like Wikipedia are much larger than comparable LSRs;
- CSRs are available for a large number of interconnected languages, including minor languages, for which LSRs might not exist.

The possible high benefit resulting from the use of CSRs in Natural Language Processing comes nonetheless with certain challenges:

- CSRs are generally less well-structured than LSRs – sometimes only semi-structured – and contain more noisy information;
- CSRs rely on community-based quality management for the assurance of accuracy and comprehensiveness, whereas LSRs typically enforce editorial quality control.

However, it should be noted that the collaborative construction approach has been argued to yield remarkable factual quality in Wikipedia (Giles, 2005), while the

[4] `http://www.openthesaurus.de/`
[5] `http://answers.yahoo.com`

quality of LSRs like WordNet has also been target of criticism (Kaplan and Schubert, 2001).

In summary, we conclude that collaboratively constructed semantic resources are promising but also challenging new resources for natural language processing. In the next section, we give a detailed overview of the main properties of linguistically constructed semantic resources, followed by an overview of collaboratively constructed semantic resources in Section 2.3.

2.2 Linguistically Constructed Semantic Resources

Linguistically constructed semantic resources (LSRs) are constructed by trained linguists following a theoretical model or corpus evidence. Examples of LSRs are WordNet (Fellbaum, 1998), GermaNet (Kunze, 2004), Cyc (Lenat and Guha, 1990), or Roget's Thesaurus (Berrey and Carruth, 1962). We focus on WordNet as the most important representative, and GermaNet as its German counterpart.

2.2.1 WordNet

WordNet (Fellbaum, 1998) is an electronic lexical database for the English language that was constructed by linguists and psycholinguists at Princeton University starting in 1985. WordNet models the English lexicon according to psycholinguistic principles (Miller, 1990; Gross and Miller, 1990; Fellbaum, 1990). Thus, it is organized according to meanings (in contrast to word forms) distinguishing it from conventional dictionaries. The major building block of WordNet are *synsets* (synonym sets), i.e. sets of lexemes that are synonyms, e.g. (*auto*, *automobile*, *car*, *machine*, *motorcar*). Each synset represents a concept and is normally linked with other synsets by semantic relations, e.g. HYPONYMY expressing an IS-A relation between two concepts like in (*car* – *jeep*). However, there are also lexical relations between word forms instead of synsets, e.g. DERIVATION between *automobile* and *automobilist*. Each synset is accompanied by a GLOSS that gives a short definition of the concept. For nouns, a gloss usually contains a reference to its hypernym and a description of how this noun differs from its hypernym. This type of definition follows a differential theory of meaning (Seco, 2005). For example, the gloss for the synset (*auto*, *automobile*, *car*, *machine*, *motorcar*) is "a motor vehicle with four wheels; usually propelled by an internal combustion engine".

Additionally, the set of synsets that a lexeme belongs to constitutes a *sense-inventory* for each lexeme. For example, the term *car* has 5 senses in WordNet. Besides the salient sense "a motor vehicle with four wheels; usually propelled by an internal combustion engine", there are other less often used senses like "the compartment that is suspended from an airship and that carries personnel and the cargo and the power plant" (*car*, *gondola*) or "a conveyance for passengers or freight on a cable railway" (*car*, *cable car*). WordNet's sense inventory has been criticized for being too fine-grained (Mihalcea and Moldovan, 2001). Especially verb senses have been found to be hard to distinguish (cf. also Appendix A.2).

Table 2.2 gives an overview of the frequency of lexical-semantic relations in WordNet 2.0. Numbers are given separately for nouns, adjectives, verbs, and adverbs.

	Noun	Adjective	Verb	Adverb
Antonymy	2074	4118	1079	722
Hypernymy	81857	-	12985	-
Hyponymy	81857	-	12985	-
Member holonymy	12205	-	-	-
Substance holonymy	787	-	-	-
Part holonymy	8636	-	-	-
Member meronymy	12205	-	-	-
Substance meronymy	787	-	-	-
Part meronymy	8636	-	-	-
Derivation	21491	-	21497	3209
Entailment	-	-	409	-
Cause	-	-	218	-
Similar to	-	22196	-	-
Pertainymy	-	4711	-	-
Also see	-	2697	597	-

Table 2.2: Selected lexical-semantic relations in WordNet 2.0, adapted from (Seco, 2005).

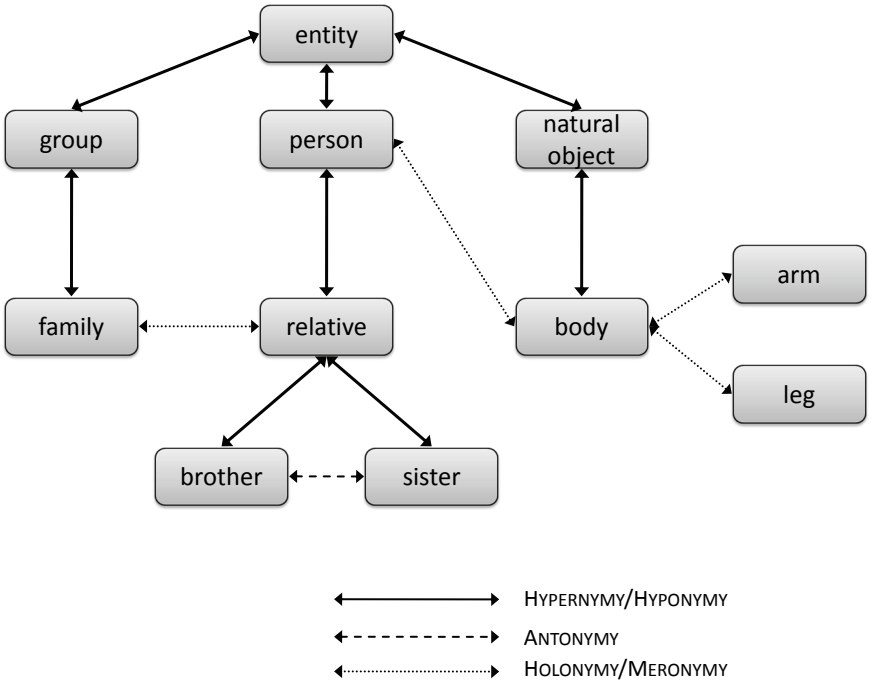


Figure 2.1: Example of nouns in WordNet organized in a taxonomic structure (adapted from (Gross and Miller, 1990)).

Each of these word classes forms a dense graph of synset nodes and relations in WordNet, while the inter-connectivity between the different word classes is very limited. However, there has been increased research in adding cross part-of-speech relations (Boyd-Graber et al., 2006).

HYPERNYMY and HYPONYMY relations are central to the organization of nouns in WordNet (see Figure 2.1). However, nouns are also often connected through HOLONYMY, MERONYMY or ANTONYMY relations. In contrast to nouns, adjectives are mainly organized using ANTONYMY and SIMILAR TO relations. This is called

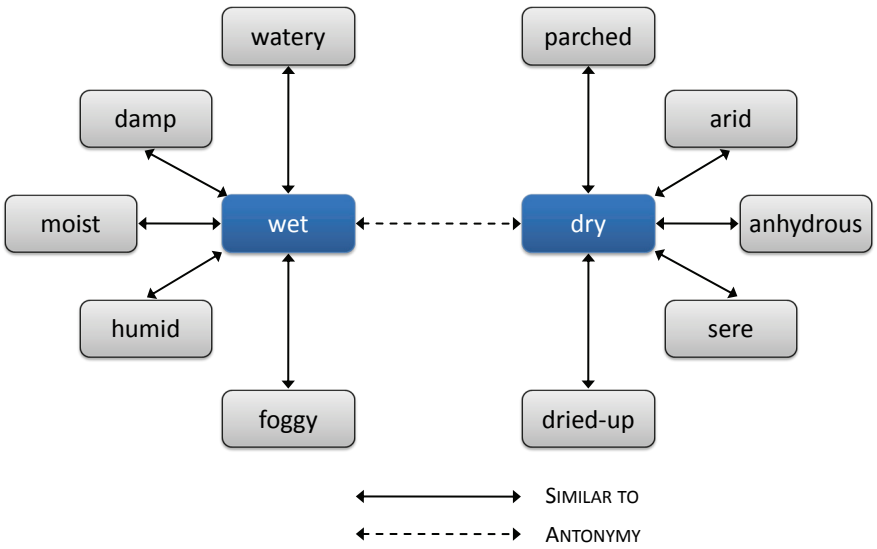


Figure 2.2: Example of adjectives in WordNet organized in a satellite approach (adapted from (Fellbaum, 1990)).

satellite approach, as similar adjectives are organized as "satellites" around a pair of adjectives in an ANTONYMY relation (see Figure 2.2).

WordNet has been used as a lexical-semantic resource for a multitude of natural language processing applications including word sense disambiguation, information retrieval, text classification, text summarization, and computing semantic relatedness. However, WordNet contains mainly common terms, while domain specific vocabulary is poorly covered. This is one of the major disadvantages when compared to CSRs like Wikipedia.

2.2.2 GermaNet

GermaNet (Kunze, 2004) is a lexical-semantic wordnet for the German language that is very similar to WordNet. However, it contains substantially less synsets than WordNet (∼50,000 compared to ∼120,000). Major differences to WordNet include (Lemnitzer and Kunze, 2002):

- the use of non-lexicalized, so called artificial concepts in GermaNet for creating well-balanced taxonomies;
- choosing a taxonomic (GermaNet) versus satellite approach (WordNet) for representing adjectives;
- the unified treatment of meronyms (GermaNet) instead of distinguishing three different types (Part, Member and Substance Meronomy in WordNet);
- cross-categorial encoding of causal relations (GermaNet), not only from verb to adjective (WordNet);
- the employment of different and more specific subcategorization frames in GermaNet

	Noun	**Adjective**	**Verb**
Lexemes	37,907	8,061	7,150
Synsets	28,107	8,855	5,171
Artificial Concepts	205	107	116
Antonymy	1,064	500	1,466
Pertonymy	13	133	1,523
Proper name	1,756	-	-
Hypernymy	31,517	9,328	5,069
Hyponymy	31,517	9,328	5,069
Holonymy	3,280	-	-
Meronymy	720	-	-
Entailment	6	8	-
Causation	24	208	1

Table 2.3: Selected lexical-semantic information in GermaNet 5.0.

These differences are also reflected in the number of lexical-semantic relations in GermaNet that are summarized in Table 2.3. For example, the taxonomic organisation of adjectives leads to a higher number of HYPERNYMY and HYPONYMY and a lower number ANTONYMY and SIMILAR TO relations between adjectives as compared to WordNet.

GermaNet and WordNet are monolingual resources. However, the wordnets of eight European languages (including English and German) have been integrated into the multilingual EuroWordNet (Vossen, 1998). EuroWordNet links synsets in different languages via an interlingual index of concepts based on WordNet. However, coverage of EuroWordNet is limited, as e.g. only 15,000 out of 50,000 German synsets have been added so far.

2.3 Collaboratively Constructed Semantic Resources

Recently, emerging Web 2.0 technologies have enabled user communities to collaboratively create new kinds of resources. Wikipedia and Wiktionary are instances of semantic resources that are constructed by mainly non-professional volunteers on the web.

2.3.1 Wikipedia

Wikipedia is a multilingual, Web based, freely available *encyclopedia*, constructed in a collaborative effort of voluntary contributors. The potential of Wikipedia as a semantic resource has recently started to get explored. It has been used in NLP tasks like text categorization (Gabrilovich and Markovitch, 2006), information extraction (Ruiz-Casado et al., 2005), information retrieval (Gurevych et al., 2007), question answering (Ahn et al., 2004), computing semantic relatedness (Zesch et al., 2007b), or named entity recognition (Bunescu and Pasca, 2006). We analyze Wikipedia as a semantic resource and compare it with conventional resources, such as dictionaries, thesauri, or semantic wordnets. We show that (i) different parts of Wikipedia reflect different aspects of these resources, and (ii) that Wikipedia contains a vast amount of knowledge about, e.g. named entities, domain specific terms, and rare word senses.

Lexical-Semantic Information

Due to an editorial decision, Wikipedia contains only terms of encyclopedic interest.[6] Thus, Wikipedia covers mainly nouns and only few adjectives and verbs. In most cases, contained verbs and adjectives redirect to their corresponding nouns, e.g. the verb *sehen* (Eng. *to see*) redirects to the phrase *Visuelle Wahrnehmung* (Eng. *visual perception*) in the German Wikipedia.

Dictionaries, thesauri, and wordnets focus on general vocabulary, while Wikipedia covers a larger number of named entities and domain specific terms, such as: *Gentest* (*DNA test*), *Makake* (*Macaque*), *Kortex* (*Cortex*), *Kompaktvan* (*Compact van*), *Nanopartikel* (*Nanoparticle*), or *Welthungerhilfe* (*German Agro Action*).

Another excellent source of lexical-semantic information in Wikipedia are article **redirects**, as they express synonymy, spelling variations and abbreviations. For example, the article about the current pope *Benedict XVI* has almost 50 redirects including spelling variations like *Pope Benedict XVI.* or *Pope Benedict 16.* Furthermore, his secular name *Joseph Ratzinger* and various combinations like *Cardinal Joseph Ratzinger* or *Joseph Alois Ratzinger*, as well as common misspellings like *Cardinal Ratsinger* are included. This example indicates the potential of Wikipedia redirects to improve named entity recognition and co-reference resolution.

The **first paragraph** of a Wikipedia article usually contains a short definition of the term the article is about. The **full article** text contains related terms and describes the meaning of the article term in detail. It may even contain translations of the article term encoded in the links to Wikipedia in other languages, turning Wikipedia into a valuable multilingual resource.

Article links point from one article to another article. Therefore, a link establishes a relation between the two terms the articles are about. Links between Wikipedia articles are untyped. Thus, they express semantic relatedness, but the type and strength of the relation are unknown. Previous work has explored the use of explicitly labeled links between articles (Völkel et al., 2006). This would turn Wikipedia into a huge semantic net, but this feature has not been added to the Wikipedia software yet.

All links between Wikipedia articles form a graph that can be used, e.g. to compute the similarity of two terms from their positions in the graph (Page et al., 1999; Jeh and Widom, 2002). On a Wikipedia HTML page, each link is visualized as a highlighted text that can be clicked. The highlighted text (called **link anchor**) does not necessarily have to be the same as the title of the article that it points to. For example, many links referring to the article with the title *Roman Empire* are actually labeled *Romans*. As a result, a link anchor may provide information about synonyms, spelling variations or related terms. Additionally, related and co-occurring terms can be extracted from a context window around a link, e.g. the link anchor *Benedict XVI.* is often preceded by *Pope*.

The **category system** in Wikipedia (Voss, 2006) can be viewed from two perspectives. From an article-centric perspective, each article can have an arbitrary number of categories, where a category is a semantic tag. From a category-centric perspective, each category can contain an arbitrary number of articles that are classified into this category. A category can have subcategories expressing MERONYMY

[6] `http://en.wikipedia.org/wiki/WP:WWIN`

Sources	Lexical-Semantic Information
Articles	
- First paragraph	Definition
- Full text	Description of meaning; related terms; translations
- Redirects	Synonymy; spelling variations, misspellings; abbreviations
- Title	Named entities; domain specific terms or senses
Article links	
- Context	Related terms; co-occurrences
- Anchor	Synonyms; spelling variations; related terms
- Target	Link graph; related terms
Categories	
- Contained articles	Semantically related terms (siblings)
- Hierarchy	Hyponymy and Mernonymy relations between terms
Disambiguation pages	
- Article links	Sense inventory

Table 2.4: Sources of lexical-semantic information in Wikipedia.

or HYPONYMY relations. For example, the category *Vehicles* has subcategories like *Military vehicles* or *Amphibious vehicles*. Thus, the categories in Wikipedia form a thesaurus. Consequently, the Wikipedia category system is called "collaborative thesaurus tagging" (Voss, 2006). Thesaurus tagging differs from collaborative tagging used by Flickr[7] or del.icio.us[8]: Tags in Wikipedia have to be chosen from the category thesaurus which is agreed upon by the community of Wikipedia users, while in collaborative tagging each user is free to define her own tags.

Wikipedia represents polysemous terms by using **disambiguation pages**. A disambiguation page lists all articles that exist for a certain term. As each article must have a unique title, articles about polysemous terms are usually differentiated by adding a disambiguation tag in parentheses, e.g. *Capital (political)* vs. *Capital (economics)*. As a result, a disambiguation page forms a sense inventory for a given term. The article without disambiguation tag is usually about the most common sense of the term, i.e. it could be used as a most-frequent-sense baseline in word sense disambiguation. However, disambiguation pages may also contain additional links to pages that do not point to a sense of the term. Therefore, extracting a sense inventory for a given term is not straightforward, as it requires to differentiate disambiguation links from other links.

Wikipedia also covers **domain specific senses** of common terms that are rarely available in LSRs. For example, *forest* has only two senses in WordNet, both related to "an area with trees". In contrast, a lot of senses are listed in Wikipedia, including a special sense denoting "data structure in computer science". Additionally, Wikipedia lists more than ten geographical entities with the name *Forest* and four famous persons with that name. Table 2.4 gives an overview of the types of lexical-semantic information found in Wikipedia.

The properties of Wikipedia can be summarized as follows: It contains a wide variety of lexical entities, but mainly nouns. Wikipedia articles cover domain specific terms and senses, but lack coverage of common concepts that are not of encyclopedic interest. Wikipedia articles express the meaning of a term by means of a short

[7] `http://www.flickr.com`
[8] `http://del.icio.us`

Language	# Articles
English	2,904,000
German	914,000
French	812,000
Polish	609,000
Japanese	591,000
Italian	574,000
Dutch	540,000
Portuguese	483,000
Spanish	480,000
Russian	400,000

Table 2.5: Size of the ten biggest Wikipedia language editions as of June 25, 2009.

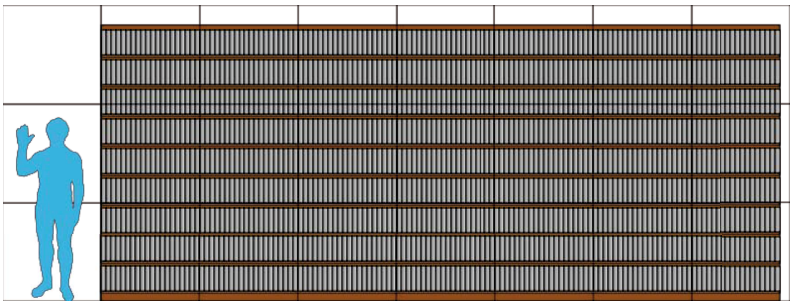


Figure 2.3: Size of the English Wikipedia (as of August 2007) visualized as standard encyclopaedic volumes on a bookshelf (adapted from `http://en.wikipedia.org/wiki/Size_of_Wikipedia`).

definition and describing text. The meaning is also implicitly expressed via the position of an article in the article graph or in the category graph. Links in Wikipedia are untyped, except for the links between categories that encode either a HYPONYMY or a MERONYMY relation. Additionally, the redirect system of Wikipedia articles can be used as a source of synonyms, spelling variations and abbreviations.

Languages and Size

Wikipedia grows rapidly, and with currently approx 9.25 million articles in more than 250 languages it has arguably become the largest collection of freely available knowledge.[9] Table 2.5 shows the number of articles in the ten biggest language editions, and Figure 2.3 visualizes the size of the English Wikipedia as standard encyclopaedic volumes on a bookshelf.

Wikipedia Mining

As Wikipedia was not designed as a semantic resource for natural language processing, much valuable lexical-semantic information is not directly available in machine readable form, but has to be extracted from Wikipedia's content or structure. For example, in contrast to WordNet or GermaNet, Wikipedia does not contain lexical-semantic relations, but only links that lack clearly defined semantics. We

[9] `http://en.wikipedia.org/wiki/Wikipedia:Size_comparisons`

call the process of extracting semantic information from different parts of Wikipedia *Wikipedia mining* and differentiate between *content mining*, *structure mining*, and *usage mining*:

Content mining refers to searching the article content for relevant knowledge. This includes, e.g. using the first paragraph as a definition, using redirects for finding spelling variations, or analyzing link labels for finding synonyms.

Structure mining refers to extracting knowledge from structural features of Wikipedia, such as the link graph or the inner structure of an article. This includes, e.g. determining the meaning of a page by means of ingoing and outgoing links on that page, or measuring the semantic similarity of two terms by computing the distance between the corresponding Wikipedia articles or categories.

Usage mining refers to extracting knowledge from revisions, usage logs or other sources reflecting the user behaviour.

In Chapter 4, we give examples of Wikipedia mining for extracting lexical-semantic information that can be used for computing semantic relatedness.

2.3.2 Wiktionary

Wiktionary is a multilingual, Web based, freely available *dictionary*, *thesaurus* and *phrase book*, designed as the lexical companion to Wikipedia. It is also collaboratively constructed by volunteers with no specialized qualifications necessary. Wiktionary targets common vocabulary and matters of language and wordsmithing. It includes terms from all parts of speech, but excludes in-depth factual and encyclopedic information, as this kind of information is contained in Wikipedia.[10] Thus, Wikipedia and Wiktionary are largely complementary.

Wiktionary has been previously applied in NLP research for sentiment classification (Chesley et al., 2006) and diachronic phonology (Bouchard et al., 2007), but has not yet been considered as a resource for computing semantic relatedness.

Lexical-Semantic Information

Although expert-made dictionaries or wordnets have been used in NLP for a long time (Wilks et al., 1990; Leacock and Chodorow, 1998), the collaboratively constructed Wiktionary differs considerably from them.

Relation types Entries in Wiktionary are accompanied with a wide range of lexical-semantic information such as part-of-speech, word sense, gloss, etymology, pronunciation, declension, examples, sample quotations, translations, collocations, derived terms, and usage notes. Lexically or semantically related terms of several types like synonyms, antonyms, hypernyms and hyponyms are included as well. On top of that, the English Wiktionary edition offers a remarkable amount of information not typically found in LSRs, including compounds, abbreviations, acronyms and initialisms, common misspellings (*basicly* vs. *basically*), simplified spelling variants (*thru* vs. *through*), contractions (*o'* vs. *of*), proverbs (*no pain, no gain*), disputed

[10] `http://en.wiktionary.org/wiki/Wiktionary:Criteria_for_inclusion`

	English Wiktionary		**German Wiktionary**	
	English	German	English	German
Entries	176,410	10,487	3,231	20,557
Nouns	99,456	6,759	2,116	13,977
Verbs	31,164	1,257	378	1,872
Adjectives	23,041	1,117	357	2,261
Examples	34,083	465	1,217	20,053
Quotations	8,849	55	0	0
Categories	4,019	992	32	89
Derived terms	43,903	944	2,319	36,259
Collocations	0	0	1,568	28,785
Synonyms	29,703	1,916	2,651	34,488
Hyponyms	94	0	390	17,103
Hypernyms	42	0	336	17,286
Antonyms	4,305	238	283	10,902

Table 2.6: The number of entries and selected types of lexical-semantic information available from the English and German editions of Wiktionary as of September 2007.

usage words (*irregardless* vs. *irrespective* or *regardless*), protologisms (*iPodian*), onomatopoeia (*grr*), or even colloquial, slang and pejorative language forms. Most of these lexical-semantic relations are explicitly encoded in the structure of a Wiktionary entry. This stands in clear contrast to Wikipedia, where links between articles usually lack clearly defined semantics. Different Wiktionary editions may include different types of information; e.g. the German edition offers mnemonics, while it currently does not contain quotations. The English edition has no collocations and only very few instances of HYPERNYMY or HYPONYMY (see Table 2.6). Like in Wikipedia, each entry in Wiktionary is additionally connected to a list of categories. Finally, entries in Wiktionary are massively linked to other entries in different ways: they are intra-linked, pointing to other entries in the same Wiktionary; they are inter-linked, pointing to corresponding entries in other language editions of Wiktionary; they also link to external semantic resources such as Wikipedia and other Web based dictionaries.

Instance structure Wiktionary allows to easily create, edit, and link HTML pages on the Web using a simple markup language. For most language editions, the user community has introduced a layout standard acting as a data schema to enforce a uniform structure of the entries. As schemas evolve over time, older entries are possibly not updated. Moreover, as no contributor is forced to follow the schema, the structure of entries is fairly inconsistent. Additionally, schemas are specific to each language edition. In LSRs, layout decisions are made in the beginning and changed only with caution afterwards. The compliance of LSR entries with the layout decisions is enforced.

Instance incompleteness Even if a Wiktionary entry follows the schema posed by a layout standard, the entry might be a stub, where most relation types are empty. Wiktionary also does not include any mechanism to enforce symmetrically defined relations (e.g. SYNONYMY) to hold in both directions. In contrast to Wiktionary, instance incompleteness is not a major concern for LSRs as new entries are usually entered along with all relevant relation types.

Quality In contrast to incompleteness and inconsistency described above, *qual-*

(a) Sizes as of February 29, 2008.

Language	Rank	Entries
French	1	730,193
English	2	682,982
Vietnamese	3	225,380
Turkish	4	185,603
Russian	5	132,386
Ido	6	128,366
Chinese	7	115,318
Greek	8	102,198
Arabic	9	95,020
Polish	10	85,494
German	12	71,399
Spanish	20	31,652

(b) Sizes as of June 25, 2009.

Language	Rank	Entries
French	1	1,392,000
English	2	1,277,000
Turkish	3	256,000
Vietnamese	4	228,000
Russian	5	215,000
Lithuanian	6	172,000
Ido	7	150,000
Greek	8	133,000
Polish	9	124,000
Chinese	10	118,000
German	17	91,000
Spanish	23	39,000

Table 2.7: Size of the largest Wiktionary language editions.

ity refers to the correctness of the encoded information itself. To our knowledge, there are no studies on the quality of the information in Wiktionary. However, the collaborative construction approach resulted in remarkable factual quality in Wikipedia (Giles, 2005), while the quality of LSRs like WordNet has also been target of criticism (Kaplan and Schubert, 2001).

Languages and Size

Wiktionary currently consists of approx 5.9 million entries in 172 language editions.[11] The size of a particular language edition of Wiktionary largely depends on how active the corresponding community is (see Table 2.7). Surprisingly, the English edition (1,277,000 entries), started on December 12, 2002, is, though the oldest, not the largest one. The French Wiktionary (1,392,000 entries), which was launched over a year later, is the largest. Other major languages like German (91,000 entries) or Spanish (39,000 entries) are not found among the top ten, while Ido, a constructed language, has the 7th largest edition of Wiktionary containing 150,000 entries.

Unlike most LSRs, each collaboratively constructed Wiktionary edition also contains entries for foreign language terms. For example, the English Wiktionary currently contains approx 10,000 entries about German words (e.g. the German term "Haus" is explained in English as meaning "house"). Additionally, corresponding entries in different languages are linked between language editions. Therefore, each language edition comprises a multilingual dictionary with a substantial amount of entries in different languages (see Table 2.6). However, as of September 2007, the English Wiktionary edition containing 176,410 English entries already exceeded the size of WordNet 3.0 containing 155,287 unique lexical units. In contrast, the German Wiktionary edition only contained about 20,000 German words in September 2007 compared to about 70,000 lexical units in GermaNet 5.0.

[11] `http://meta.wikimedia.org/wiki/Wiktionary`

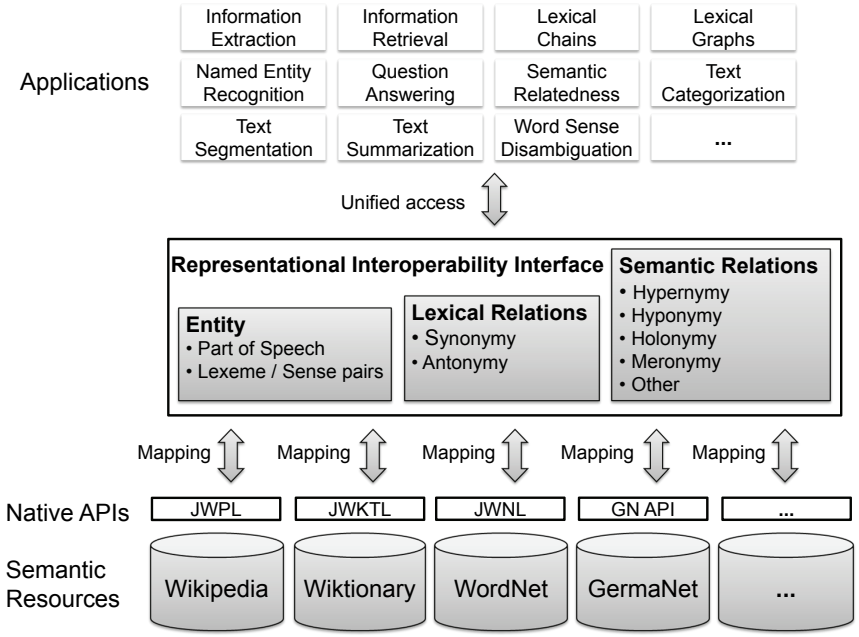


Figure 2.4: System architecture enabling representational interoperability.

2.4 Interoperability of Semantic Resources

Without collaboratively constructed semantic resources being available, natural language processing applications usually only had to deal with one semantic resource and were tailored towards it. With a wide range of semantic resources available, it would be necessary to adjust applications to each semantic resource. We address this problem by developing a model of representational interoperability between semantic resources, which abstracts over the differences in their structure and enables a uniform representation of their content in terms of *Entities* and lexical-semantic *Relations* between them. *Entities* consist of a set of lexeme–sense pairs along with a part-of-speech (PoS). The currently supported relations are the lexical relations Synonymy and Antonymy and the semantic relations Hypernymy, Hyponymy, Co-Hyponymy, Holonymy, Meronymy and Other. NLP algorithms can then be implemented in a one-time effort, as they only have to know about generalized *Entities* and *Relations* instead of being adapted to each semantic resource. Currently, we have integrated the LSRs WordNet, GermaNet, Cyc (Lenat and Guha, 1990), Roget's Thesaurus (Jarmasz and Szpakowicz, 2003), Leipzig Annotation Project (Biemann, 2005), and the CSRs Wikipedia and Wiktionary.

The system architecture enabling representational interoperability of semantic resources is presented in Figure 2.4. Each semantic resource implements the generic interface by means of their native application programming interfaces (APIs). As entries and relations are modeled differently in each semantic resource, they are mapped onto *Entities* and *Relations*. For example, a synset from WordNet is mapped to an *Entity* by adding each synonym from the synset as a lexeme in the *Entity* together with its sense number and its PoS. Likewise, an article from the Wikipedia is mapped to an *Entity* by adding the article name and all redirects as lexemes. In this case, sense and PoS are left unspecified, as this information cannot be di-

rectly retrieved from Wikipedia. Similarly, the encoded relations between WordNet synsets or Wikipedia articles are mapped onto the given set of lexical-semantic relations. Additional information originally related to the concepts, such as glosses or examples, does not belong to the representation of an *Entity*, but remains programmatically accessible through the interface.

To our knowledge, there is no other framework aiming at representational interoperability between LSRs and CSRs. Related work has focused on combining semantic resources on the content level in order to produce enriched semantic resources of greater coverage by merging or mapping concepts (Fröhner et al., 2005; Shi and Mihalcea, 2005; Suchanek et al., 2007; Medelyan and Legg, 2008).

It is obvious that this generalized model cannot support the same level of expressiveness as directly accessing a semantic resource. Additionally, the generalization comes with some caveats: For example, the articles of Wikipedia that redirect to each other are treated as lexemes of the same concept. However, redirects can also be spelling variants or common misspellings that we assume to be in a near-SYNONYMY relation with each other. Furthermore, synonyms listed in Wiktionary are currently not integrated as lexemes of an entity, as is the case e.g. for WordNet. This is due to the fact that in Wiktionary SYNONYMY is not necessarily a symmetric relation. However, we believe that these problems are compensated by the advantages of the representational interoperability:

- Each NLP application has to be implemented only once and can then be applied to all semantic resources;
- experimental results obtained using different semantic resources are better comparable;
- the representational interoperability framework provides a basis for further work on full interoperability (including content alignment) of CSRs and LSRs.

2.5 Chapter Summary

We described linguistically and collaboratively constructed semantic resources that can be used as knowledge sources for a wide range of NLP applications. We focused on the emerging resources Wikipedia and Wiktionary that are not primarily intended for usage in NLP. However, a detailed analysis of their properties unveils that they contain a wide range of lexical-semantic information. We presented a representational interoperability framework that is used to access all semantic resources in a uniform manner. In the remainder of this thesis, we are going to explore the potential of collaboratively constructed semantic resources as knowledge sources for computing semantic relatedness. In the next chapter, we describe the state of the art in semantic relatedness measures, and in Chapter 4, we focus on the adaptation of semantic relatedness measures to collaboratively constructed semantic resources.

Chapter 3

Semantic Relatedness

Computing the semantic relatedness between words is a pervasive task in natural language processing with applications in word sense disambiguation (Patwardhan and Pedersen, 2006), semantic information retrieval (Gurevych et al., 2007), or information extraction (Stevenson and Greenwood, 2005). Humans can easily judge the semantic relatedness between two words. For example, they can easily tell that *car* and *drive* are strongly related, while there is no such strong connection between *car* and *eat*. This human ability is backed by their experience and knowledge, which makes it a hard task for machines. If a machine should solve this task, it also needs some knowledge source. Usually, this knowledge comes from (i) large text collections (called *corpora*) or from (ii) semantic resources (as described in Chapter 2). In this thesis, we focus on the latter one, i.e. algorithms using knowledge derived from semantic resources.[1]

In this chapter, we first define semantic relatedness more formally in Section 3.1. In Section 3.2, we describe the state of the art in computing semantic relatedness, and categorize existing algorithms into four distinct types of measures, where each type uses different properties of the underlying semantic resources.

3.1 Definition and Terminology

We formally define *semantic relatedness* as the strength of a relation between two concepts:

$$rel(c_1, c_2) \in [0, 1]$$

The strength of the relation expresses the degree of relatedness between the two concepts. A value of 0 means that the two concepts have absolutely nothing in common, and a value of 1 means that the two concepts are identical. Semantic relatedness is symmetric, i.e. $rel(c_1, c_2) = rel(c_2, c_1)$.

Algorithms that quantify the strength of the relation between two concepts are called *semantic relatedness measures*.

[1] For an overview on distributional approaches we refer to (Weeds and Weir, 2005). Distributional methods have recently shown to yield competitive performance on some evaluation datasets. The interested reader may refer to (Mohammad et al., 2007).

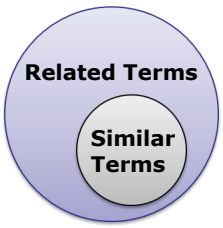


Figure 3.1: Relationship between the set of similar terms and the set of related terms.

3.1.1 Semantic Relatedness vs. Semantic Similarity

Semantic similarity is typically defined via the lexical relations of SYNONYMY (*automobile – car*) and HYPERNYMY/HYPONYMY (*vehicle – car*). However, dissimilar words can also be semantically related, e.g. via functional relationships (*night – dark*) or when they are antonyms (*high – low*). Another limitation is that similarity is only defined between terms of the same part-of-speech. Many NLP applications require knowledge that goes beyond similarity (Budanitsky and Hirst, 2006). Thus, **semantic relatedness** is defined to cover any kind of lexical or functional association that may exist between two words (Budanitsky and Hirst, 2006). Semantic relatedness determines whether two words are in some way related, even if they are not similar or have different parts-of-speech. Consequently, semantic similarity is a special case of the broader defined semantic relatedness, i.e. two words that are similar are also related, but the inverse is not true (see Figure 3.1).

3.1.2 Semantic Relatedness vs. Semantic Distance

Semantic distance is the inverse of both semantic relatedness and semantic similarity. Thus, the term may cause confusion, as it can be used when talking about either just similarity or relatedness (Budanitsky and Hirst, 2006). For example, antonymous concepts like "high" and "low" are dissimilar and hence *distant* in terms of semantic similarity, but are related and thus *close* in terms of semantic relatedness.

Semantic distance measures can always be transformed into a semantic similarity or a semantic relatedness measure. However, depending on the specific algorithm, one of the representations is more 'natural' and thus much easier to understand. For example, an algorithm that determines the distance of two concepts in a graph representation of a semantic resource is more naturally represented as a distance measure than a relatedness measure. However, it can easily be transformed into a relatedness measure by subtracting the distance from the maximum possible path length in the graph.

3.2 Semantic Relatedness Measures

A multitude of semantic relatedness measures working on structured semantic resources have been proposed. Figure 3.2 gives an overview of the development of

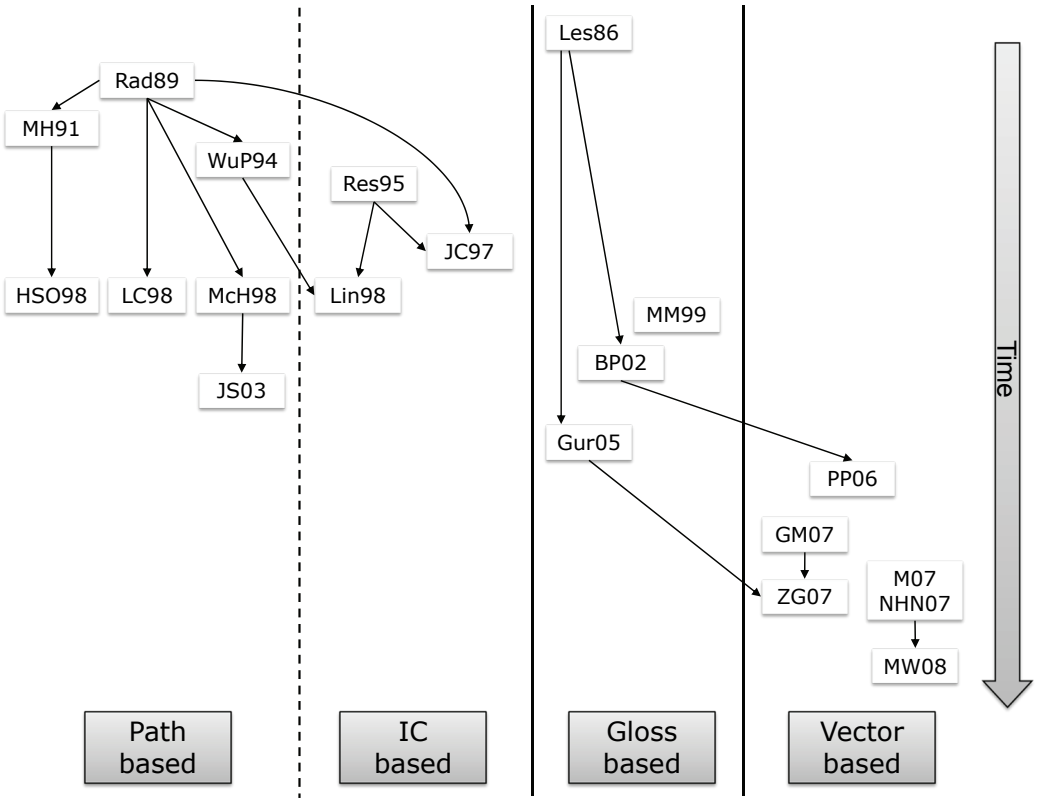


Figure 3.2: Chronological development of semantic relatedness measures.

semantic relatedness measures on a historic time scale. Measures can be categorized into:

Path based measures relying on paths in a graph of concepts (Rada et al., 1989; Wu and Palmer, 1994; Hirst and St-Onge, 1998; McHale, 1998; Leacock and Chodorow, 1998; Jarmasz and Szpakowicz, 2003)

Information content based measures taking into account the information content of a concept (Resnik, 1995; Jiang and Conrath, 1997; Lin, 1998)

Gloss based measures relying on term overlaps between definitions of concepts (Lesk, 1986; Mihalcea and Moldovan, 1999; Banerjee and Pedersen, 2002; Gurevych, 2005)

Vector based measures constructing a vector representation of a concept (Patwardhan and Pedersen, 2006; Gabrilovich and Markovitch, 2007; Milne, 2007; Nakayama et al., 2007; Milne and Witten, 2008a)

The figure shows the recent shift from path based and information content based measures (using only explicitly modelled knowledge) to gloss based and vector based measures which are able to use information that is not explicitly modelled as a relation in a semantic resource, as it is drawn from the description of a concept. In the remainder of this section, we describe all the measures in the different categories in more detail.

3.2.1 Path Based Measures

Path based measures determine the length of the path between nodes representing concepts in a semantic resource (e.g. a wordnet, a thesaurus, or the Wikipedia category graph). The shorter the path, the higher the relatedness between concepts.

Rada et al. (1989) use the path length l between two nodes to compute semantic relatedness. This measure (abbreviated as **Rad89**) can either be a semantic similarity or a semantic relatedness measure depending on the type of edges that are allowed in a path. For example, if only edges corresponding to classical lexical-semantic relations are allowed, Rad89 is a semantic similarity measure. However, if also edges corresponding to non-classical relations are allowed, it is a semantic relatedness measure. Rad89 can be computed as follows:

$$dist_{\text{Rad89}}(c_1, c_2) = l(c_1, c_2)$$

where *dist* means that the measure returns a distance value instead of a relatedness value, and $l(c_1, c_2)$ returns the number of edges on the path from c_1 to c_2. The distance value can be easily transformed into a relatedness value by subtracting it from the maximum path length of the graph, $rel_{\text{Rad89}}(c_1, c_2) = l_{max} - l(c_1, c_2)$.

Jarmasz and Szpakowicz (2003) (**JS03**) adapt the Rad89 measure to Roget's thesaurus as a semantic resource based on the work of McHale (1998) (**McH98**). JS03 is also a relatedness measure as the relations in Roget's thesaurus are not restricted to classical lexical-semantic relations.

As polysemous words may have more than one corresponding concept in a semantic resource, the resulting semantic relatedness score between two words w_1 and w_2 can be calculated as

$$rel = \begin{cases} \min\limits_{c_1 \in C(w_1), c_2 \in C(w_2)} dist(c_1, c_2) \\ \max\limits_{c_1 \in C(w_1), c_2 \in C(w_2)} rel(c_1, c_2) \end{cases}$$

where $C(w_i)$ is the set of concepts that represent senses of word w_i. That means, the relatedness of two words is equal to that of the most related (least distant) pair of concept nodes, depending on whether the measure returns a relatedness $rel(c_1, c_2)$ or a distance $dist(c_1, c_2)$ value.

Leacock and Chodorow (1998) **LC98** normalize the path length with the depth of the graph:

$$sim_{\text{LC98}}(c_1, c_2) = -\log \frac{l(c_1, c_2) + 1}{2 \cdot depth}$$

where *depth* is the length of the longest path from the root node of the taxonomy to a leaf node. The prefix *sim* means that the measure is a similarity measure in its original definition, as it was defined on WordNet using only taxonomic links. The scaling factor $2 \cdot depth$ assumes a tree-like structure, where the longest possible path runs from a leaf to the root node and back to another leaf node. Note that in contrast to the original definition, we increase the path length by 1 as by definition $l(c_i, c_i)$ returns 0, and $log(0)$ is undefined.

The length of the path $l(c_1, c_2)$ was originally measured in nodes, however we redefined it to counting edges. It is a controversial question whether to count edges

or nodes in a path. Both approaches have been used in the past. We recommend counting edges based on the following argumentation: If two concepts are identical, they are mapped to the same node. Both methods, measuring distance in edges as well as in nodes, will assign a distance of 0 in this case. If two nodes are direct neighbors, they are one edge but zero nodes apart. As a result, when measuring in nodes, there is no way to differentiate the distance of identical or neighboring nodes. This clearly puts measuring in edges in favor. Thus, we have redefined the original definitions of measures slightly from measuring nodes to edges, wherever necessary.

The simple path length methods described above do not take into account that concepts higher in the taxonomy are more abstract, i.e. that a path with a length of 1 between abstract concepts near the top of the taxonomy should yield a lower similarity value than a path of the same length between specific concepts on the leaf level of the taxonomy. Many measures have been proposed to overcome this limitation. For example, Wu and Palmer (1994) introduce a measure (**WuP94**) that uses the notion of a lowest common subsumer of two concepts $lcs(c_1, c_2)$. An *lcs* is the first shared concept on the paths from the concepts to the root concept of the hierarchy.

$$sim_{\text{WuP94}} = \frac{2 \cdot depth(lcs)}{l(c_1, lcs) + l(c_2, lcs) + 2 \cdot depth(lcs)}$$

WuP94 is a similarity measure, as an *lcs* is only defined in a taxonomy.

Hirst and St-Onge (1998) adapt a measure (**HSO98**) originally described by Morris and Hirst (1991) (**MH91**) to work with WordNet instead of Roget's Thesaurus (Berrey and Carruth, 1962). Using the HSO98 measure, two words have the highest relatedness score, if (i) they are in the same synset, (ii) they are antonyms, or (iii) one of the words is part of the other (e.g. *car* and *car park*). In all other cases, relatedness depends on the path between the concepts, where long paths and direction changes (upwards, downwards, horizontally) are penalized. The resulting formula is

$$rel_{\text{HSO98}}(c_1, c_2) = C - len(c_1, c_2) - k \cdot turns(c_1, c_2)$$

where C and k are constants, *len* is the length of the path and *turns* counts the number of direction changes in the path. The HSO98 measure is a relatedness measure as paths are not restricted to taxonomic links.

3.2.2 Information Content Based Measures

Information content (IC) approaches are based on the assumption that the similarity of two concepts can be measured by the amount of information they share. IC describes how informative a term is. Intuitively, naming an entity a *Porsche* tells you more about its properties than naming it just *car*. The more specific a term, the higher its information content. The specificity of a term can be measured using its corpus frequency or its position in a taxonomy.

In a taxonomy, the shared properties of two concepts are expressed by their lowest common subsumer *lcs*. Consequently, Resnik (1995) defines semantic similarity (**Res95**) between two nodes as the information content value of their *lcs*:

$$sim_{\text{Res95}}(c_1, c_2) = IC_{Res95}(lcs(c_1, c_2))$$

The information content of a concept can be computed as

$$IC_{Res95}(c) = -\log p(c)$$

where $p(c)$ is the probability of encountering an instance of c in a corpus. The probability $p(c)$ can be estimated from the relative corpus frequency of c and the probabilities of all concepts that c subsumes (Resnik, 1995). This definition of IC is bound to the availability of a large corpus, and the obtained IC values are relative to that corpus. Hence, Seco and Hayes (2004) introduce the *intrinsic information content* (IIC) which is computed only from structural information of the taxonomy and yields better results on some English datasets. It is defined as:

$$IIC_{Sec04}(c) = 1 - \frac{\log(hypo(c_i) + 1)}{\log(|C|)}$$

where $hypo(c_i)$ is the number of all hyponyms of a concept c_i and $|C|$ is the total number of concepts in the taxonomy. Intrinsic IC is equivalent to Resnik's definition of IC if we set the corpus frequency of each word to 1, and a word's frequency count is not divided between its concepts. Both definitions of IC yield similar results, indicating that ignoring the frequency information does not result in a performance loss. The depth scaling effect used by both definitions of IC seems to be more important than the frequency scaling.

Jiang and Conrath (1997) define a measure (**JC97**) derived from Res95 by additionally using the information content of the concepts. The original formula returns a distance value, but can be easily transformed to return a relatedness value instead.

$$dist_{\text{JC97}}(c_1, c_2) = IC_{Res95}(c_1) + IC_{Res95}(c_2) - 2 \cdot IC_{Res95}(lcs)$$

$$sim_{\text{JC97}}(c_1, c_2) = 2 - IC_{Res95}(c_1) - IC_{Res95}(c_2) + 2 \cdot IC_{Res95}(lcs)$$

Lin (1998) defines a universal measure (**Lin98**) derived from information theory.

$$sim_{\text{Lin98}}(c_1, c_2) = 2 \cdot \frac{IC_{Res95}(lcs)}{IC_{Res95}(c_1) + IC_{Res95}(c_2)}$$

IC based semantic relatedness measures traditionally form a category of semantic relatedness measures distinct from path based measures, as they were originally defined using distributional determined IC values. However, when using the intrinsic information content (Seco and Hayes, 2004) that is derived from the taxonomic link structure of a semantic resource, the distinction between path based and IC based semantic relatedness measures becomes blurred. Thus, we treat them as a single measure type in the remainder of this thesis.

3.2.3 Gloss Based Measures

Dictionaries or wordnets usually contain short glosses for each concept that are used by gloss based measures to determine the relatedness of concepts.

Lesk (1986) introduces a measure (**Les86**) based on the amount of word overlap in the glosses of two concepts, where higher overlap means that two terms are more related:

$$rel_{\text{Les86}}(c_1, c_2) = |gloss(c_1) \cap gloss(c_2)|$$

where $gloss(c_i)$ returns the multiset of words in a concept's gloss.

Banerjee and Pedersen (2002) propose a more sophisticated text overlap measure (**BP02**) that additionally takes into account the glosses of related concepts forming an extended gloss *extGloss*. This overcomes the problem that glosses in WordNet are very short. The measure is defined as:

$$rel_{\text{BP02}}(c_1, c_2) = |extGloss(c_1) \cap extGloss(c_2)|$$

where $extGloss(c_i)$ returns the multiset of content words in the extended gloss.

Mihalcea and Moldovan (1999) (**MM99**) take a similar approach in their word sense disambiguation system. They construct a linguistic context for each noun or verb sense c_i by concatenating the nouns found in the glosses of all WordNet synsets in the sub-hierarchy of c_i. The relatedness between two senses is then computed as the number of overlapping nouns in the corresponding contexts.

Gloss based measures cannot be directly used with semantic resources like GermaNet that do not contain textual definitions of concepts. Therefore, Gurevych (2005) constructed pseudo glosses (**Gur05**) by concatenating concepts that are in close relation (SYNONYMY, HYPERNYMY, MERONYMY, etc.) to the original concept. This is based on the observation that most content words in glosses are in close relation to the described concept. For example, the pseudo gloss for the concept *tree (plant)* would be:

> woody plant, ligneous plant, stump, tree stump, crown, treetop, limb, tree branch, trunk, tree trunk, bole, burl, ...

showing a high overlap with its WordNet gloss:

> a tall perennial *woody plant* having a main *trunk* and *branches* forming a distinct elevated *crown*.

The measure can be formalized as follows:

$$rel_{\text{Gur05}}(c_1, c_2) = |pseudoGloss(c_1) \cap pseudoGloss(c_2)|$$

where $pseudoGloss(c_i)$ returns the multiset of content words in the pseudo gloss.

3.2.4 Vector Based Measures

In this section, we focus only on semantic relatedness measures where concept vectors are derived from a semantic resource, rather than on distributional vectors derived from co-occurrence counts. Thus, we use the term *vector based measure* interchangeably with *concept vector based measure*.

Patwardhan and Pedersen (2006) represent a concept by a second order gloss vector using WordNet glosses (**PP06**). They start with first order context vectors, i.e. a vector of co-occurrence counts for each content word in a corpus. In this case, the corpus is the set of WordNet glosses. A second order gloss vector $glossVector(c_i)$ is then constructed from the gloss of the target concept c_i by combining the first order gloss vectors of words that appear in that gloss. For example, from the gloss of *tree (plant) "a tall perennial woody plant having a main trunk and branches forming a distinct elevated crown"*, the algorithm would construct a second order gloss

vector from the first order gloss vectors of *plant*, *trunk*, *branches*, *crown*, etc. The relatedness of two concepts is then computed as the cosine of the second order gloss vectors.[2]

$$rel_{\text{PP06}}(c_1, c_2) = \frac{glossVector(c_1) * glossVector(c_2)}{|glossVector(c_1)|\,|glossVector(c_2)|}$$

Patwardhan and Pedersen also introduce a variant of this algorithm to compensate for short glosses, where a gloss is augmented with glosses of concepts that are in close relation to the original concept. This is conceptually close to the extended BP02 measure described in the previous section.

Gabrilovich and Markovitch (2007) introduce another vector based measure, where the meaning of a word w is represented as a high dimensional concept vector $\vec{d}(w) = (d_1, \ldots, d_N)$. Each element d_i represents a document, and the value of d_i depends on the occurrence of the word w in this document. This is very similar to the approach by Qiu and Frei (1993) for constructing a similarity thesaurus used in information retrieval.

Gabrilovich and Markovitch derive the concept vector from Wikipedia articles $a_1, \ldots, a_N$, as each article focuses on a certain topic, and can thus be viewed as expressing a concept. The dimension of the concept vector is the number of Wikipedia articles N. Each element of the concept vector $\vec{d}$ is associated with a certain Wikipedia article a_i. If the word w can be found in this article, the word's tf.idf score (Salton and McGill, 1983) in the article a_i is assigned to the concept vector element d_i. Otherwise, 0 is assigned.

$$d_i = \begin{cases} tf.idf(w), & w \in a_i \\ 0, & otherwise \end{cases}$$

As a result, the vector $\vec{d}(w)$ represents the word w in concept space. Semantic relatedness of two words can then be computed as the cosine of their corresponding concept vectors:

$$rel_{\text{GM07}}(w_1, w_2) = \frac{\vec{d}(w_1) * \vec{d}(w_2)}{\left|\vec{d}(w_1)\right| \left|\vec{d}(w_2)\right|}$$

Milne (2007) (**M07**) and Nakayama et al. (2007) **NHN07** in parallel introduced a vector based measure that is specific to Wikipedia as it relies on heavy linking between articles that cannot be found in other semantic resources. They use links in Wikipedia articles that point to other articles (called 'targets'). The more targets two articles share, the higher their semantic relatedness. Links to targets are considered less significant if many other articles also link to the same target. For example, a link to a very common target like *automobile* is less important than a link to a more specific target like *Ethanol fuel*. Formally, the weight ω of a link between a source s and a target t is defined as:

$$\omega(s \to t) = \begin{cases} \log\left(\frac{N}{|T|}\right), & s \in T \\ 0, & otherwise \end{cases}$$

[2] This measure displays properties of both gloss based and vector based approaches. It is categorized as a vector based measure, because the final relatedness computation relies on a vector representation that is only derived from glosses.

where T is the set all articles that link to t, and N is the number of Wikipedia articles. An article is then represented as a vector $\vec{l}$ of weighted outgoing links l. The semantic relatedness of two terms is computed as the cosine of the link weight vectors of the corresponding articles.

$$rel_{\text{M07/NHN07}}(a_1, a_2) = \frac{\vec{l}(a_1) * \vec{l}(a_2)}{\left|\vec{l}(a_1)\right| \left|\vec{l}(a_2)\right|}$$

where a_i are Wikipedia articles corresponding to terms. An article corresponds to a term if its title or one of its redirects matches the term, or if the article is linked on a disambiguation page whose title matches the term.

Milne and Witten (2008a) **MW08** present a refined version of their measure using the Wikipedia link structure by taking also incoming links into account. The measure is modeled after the *Normalized Google Distance* (Cilibrasi and Vitanyi, 2007), but takes link co-occurrences instead of term co-occurrences into account. It is formally defined as:

$$rel_{\text{MW08}}(a_1, a_2) = \frac{\log\left(\max\left(|A_1|, |A_2|\right)\right) - \log\left(|A_1 \cap A_2|\right)}{log\left(N\right) - \log\left(\min\left(|A_1|, |A_2|\right)\right)}$$

where a_i are Wikipedia articles and A_i are the sets of articles that link to a_i. N is the number of Wikipedia articles. They also refine the process of selecting corresponding articles for a given term by taking link anchors into account.

3.3 Chapter Summary

In this chapter, we introduced semantic relatedness, a central concept in NLP with a multitude of important applications. We categorized semantic relatedness measures into four categories differing in their usage of the underlying semantic resource. In the next chapter, we focus on adapting these measures, which were mostly defined on linguistically constructed semantic resources, to collaboratively constructed semantic resources.

Chapter 4

Adapting Semantic Relatedness Measures

Many semantic relatedness measures introduced in Chapter 3 have been defined on a certain type of semantic resource, e.g. the gloss based measure by Lesk (1986) on dictionaries, or the concept vector based measure by Gabrilovich and Markovitch (2007) on Wikipedia. However, most measures can be adapted to other semantic resources. Due to structural or semantic differences between semantic resources, an adaptation process might be necessary that is described in this chapter. In Section 4.1, we focus on adapting path and information content based measures to collaboratively constructed semantic resources. For this purpose, we also perform a graph-theoretic analysis of semantic resources. Section 4.2 describes the adaptations necessary for gloss based measures, and Section 4.3 focuses on the generalization of vector based measures from Wikipedia to other semantic resources.

4.1 Path and IC Based Measures

Path based measures rely on paths in a graph of concepts, where nodes represent concepts, and edges represent relations between these concepts. For most semantic resources, this graph can be easily constructed. For example in a dictionary, a term's definition contains other terms that can be found in the dictionary. This can be used to form a relationship graph between dictionary entries. Thesauri consist of a hierarchy of categories, where related terms are grouped together on the leaf level. Semantic wordnets group synonyms together (*synsets*) and link them by means of semantic relations between these *synsets* or lexical relations between single lexemes. The result is a graph with a backbone consisting of classical taxonomic relations.

In the case of Wikipedia, things are more difficult, as Wikipedia articles are not organized in a hierarchical structure as required by most path based and all IC based measures. This role is filled by the *Wikipedia category graph* (**WCG** for short). This graph is known to resemble a thesaurus, where relations between categories are not as strictly defined as in linguistically motivated wordnets, but their semantics is more of the kind "broader term" or "narrower term" (Voss, 2006). However, the WCG alone is not sufficient to compute the semantic relatedness between terms, as its nodes usually represent generalized concepts or categories instead of simple

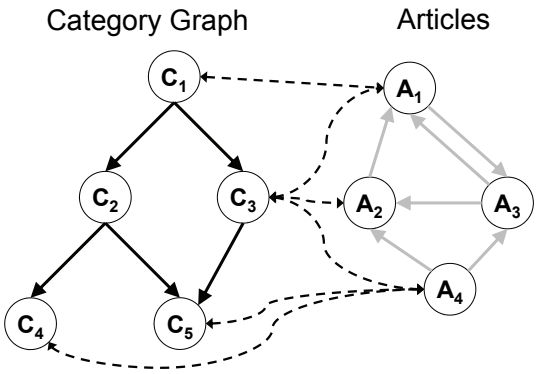


Figure 4.1: Relations between Wikipedia articles and categories in the category graph.

terms. In ontological terms, it does not contain any instances, but only classes. For example, the English WCG does not contain a node for "seat belt" but a category "Automotive safety technologies". Therefore, the WCG alone would not provide sufficient coverage for our experiments (Zesch et al., 2007b; Zesch and Gurevych, 2010). Thus, we use the mutual links between Wikipedia articles and categories (see Figure 4.1) to connect articles to categories. While the number of categories in Wikipedia (about 210,000 for the version from February 6th, 2007 used in this thesis) is comparable to the number of lexical units in WordNet (about 150,000 in WordNet 3.0), the number of articles and redirects in Wikipedia is an order of magnitude higher (3,300,000) providing much more coverage. In the remainder of this section, we give a formal description of the adaptation process.

4.1.1 Adapting to Wikipedia

To compute semantic relatedness of two words w_1 and w_2 using Wikipedia, we first retrieve the articles or disambiguation pages with titles that equal w_1 and w_2 (see left side of Figure 4.2). If we hit a redirect page, we retrieve the corresponding article or disambiguation page instead. In case of an article, we insert it into the candidate article set (A_1 for $w1$, A_2 for w_2). In case of a disambiguation page, the page contains links to all encoded word senses, but it may also contain other links. Therefore, we only consider links conforming to the pattern ⟨Title (DisambiguationText)⟩ where '(DisambiguationText)' is optional – (e.g. "Bank" or "Bank (sea floor)"). Following all such links gives the candidate article set. If no disambiguation links conforming to the pattern are found, we take the first link on the page, as most important links tend to come first, and add the corresponding article to the candidate set. We form pairs from each candidate article $a_i \in A_1$ and each article $a_j \in A_2$. We then compute $rel(a_i, a_j)$ for each pair. The output of the algorithm is the maximum semantic relatedness value

$$rel(w_1, w_2) = \max_{a_i \in A_1, a_j \in A_2} (rel(a_i, a_j))$$

For computing the semantic relatedness value $rel(a_i, a_j)$, we define C_1 and C_2 as the set of categories assigned to article a_i and a_j, respectively (see right side of

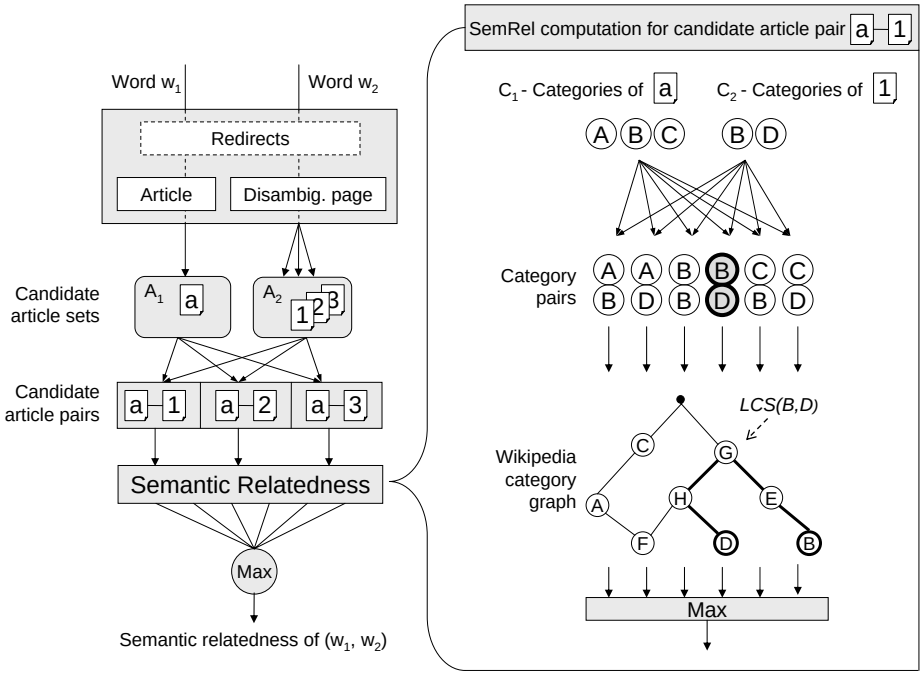


Figure 4.2: Adaptation of path based and IC based semantic relatedness measures to Wikipedia.

Figure 4.2). We then determine the semantic relatedness value for each category pair (c_k, c_l) with $c_k \in C_1$ and $c_l \in C_2$. We choose the maximum semantic relatedness value among all pairs (c_k, c_l).

For the information content based measures, we need to compute the information content of a concept. We use intrinsic information content (IC_{Sec04}), relying on hypernym counts in the original definition. As links in the WCG are untyped, we define all 'narrower term' links (i.e. links to concepts deeper in the hierarchy) as pseudo hyponyms. Efficiently counting the hyponyms of a node requires to break cycles that may occur in the WCG. We perform a colored depth-first-search to detect cycles, and break them as visualized in Figure 4.3. A link pointing back to a node closer to the root node is deleted, as it violates the rule that downward links in the WCG typically express 'narrower term' relations. If the cycle occurs between nodes on the same level, we cannot decide based on that rule and randomly delete one of the links running on the same level. This strategy never disconnects any nodes from the graph.

Strube and Ponzetto (2006) take a similar approach to adapting some WordNet based measures to Wikipedia using the category graph. However, they use a different disambiguation heuristic. It relies on finding a common substring in links on disambiguation pages. As there is no Wikipedia editing principle that enforces a standardized vocabulary, this strategy sometimes fails even for closely related concepts (e.g. "Bank (sea floor)" and "Water" do not have any common substring). We also found that the substring heuristic is used in less than 5% of cases. Thus, the disambiguation strategy almost fully relies on a fallback strategy that takes the first sense on a page that is considered to be the most common one. This strategy is not optimal for modelling the lexical cohesion of a text, as cohesion might be es-

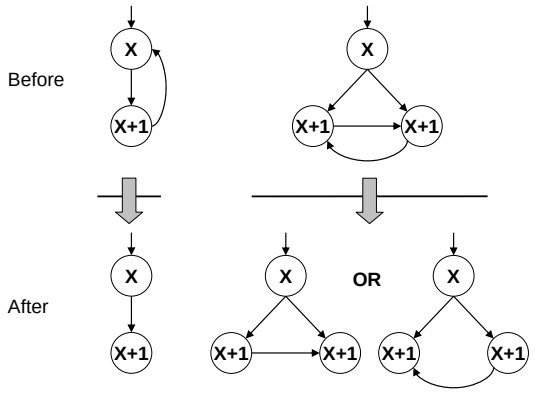


Figure 4.3: Breaking cycles in the WCG.

tablished by rare senses in certain domains. For example, in the sentence "This tree spans the whole graph." the special sense of 'tree (graph theory)' contributes much to the lexical cohesion of the sentence, but cannot be determined using a heuristic that depends on the most common sense. However, better disambiguation strategies have to be developed, e.g. incorporating contextual information.

Walkthrough example If we want to determine the semantic relatedness between the terms "Zuse" (the last name of a famous computer pioneer) and "Dijkstra" (the last name of another famous computer scientist), we first retrieve the articles corresponding to these terms. For "Zuse", we get redirected to the article "Konrad Zuse". For "Dijkstra", we hit a disambiguation page, as there are a couple of famous persons called "Dijkstra". For the sake of the example, we consider only the first two persons mentioned on the disambiguation page "Edsger W. Dijkstra" and "Rineke Dijkstra". We now form article pairs between the one article related to "Zuse" and each of the two persons called "Dijkstra" yielding "Zuse/Edsger W. Dijkstra" and "Zuse/Rineke Dijkstra". Then, we look at the categories assigned to each article. For the sake of the example, we only consider one category per article. The articles "Zuse" and "Edsger W. Dijkstra" both have the category *Computer pioneers*, while "Rineke Dijkstra" has the category *Dutch photographers*. We now form category pairs yielding *Computer pioneers/Computer pioneers* and *Computer pioneers/Dutch photographers*. We then measure the semantic relatedness in terms of the path length between the categories in the hierarchy (Rad98 measure). Let the path length between *Computer pioneers/Dutch photographers* be 4. The path length for the identical categories in *Computer pioneers/Computer pioneers* is always 0. If the maximum path length in the category graph is 8, then the semantic relatedness between *Computer pioneers/Dutch photographers* is 0.5, and between *Computer pioneers/Computer pioneers* it is 1. We now take the maximum of those values (that is 1) and select it as the value of semantic relatedness between "Zuse" and "Dijkstra", i.e. both terms are very highly related. Note that we could not have used the category graph alone for computing semantic relatedness, as it contains neither a node "Zuse" nor a node "Dijkstra".

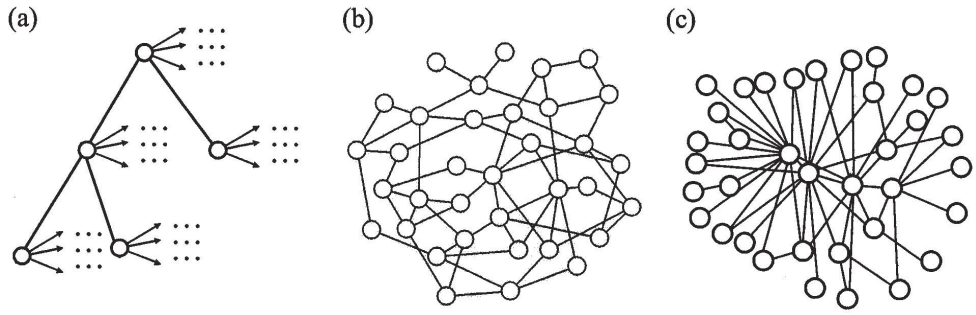


Figure 4.4: Structures of semantic networks according to (Steyvers and Tenenbaum, 2005): a) a taxonomy, b) an arbitrary graph, c) scale-free, small-world graph.

4.1.2 Adapting to Wiktionary

The adaptation process for Wiktionary is more straightforward than for Wikipedia, as Wiktionary is much more similar to a classical wordnet or dictionary. However, as we showed in Section 2.3.2, the English Wiktionary does not contain taxonomic links due to a decision by the user community. Thus, path and IC based measures relying on taxonomic links cannot be adapted to the English Wiktionary. Other Wiktionary language editions (e.g. German) that contain taxonomic links can directly be used as a semantic resource for path and IC based measures without any adaptation. Path based measures that do not rely on taxonomic links (e.g. simple path length (Rada et al., 1989)) can be directly applied to all Wiktionary language editions.

4.1.3 Graph-Theoretic Analysis of Semantic Resources

So far, we have only considered whether a semantic resource contains a graph structure that can be used for computing semantic relatedness based on paths or information content. However, the properties of a graph might differ among resources and the obtained results might be rendered invalid. Thus, a graph-theoretic analysis of Wikipedia is required to determine, whether graph based semantic relatedness measures developed for semantic wordnets can be applied to it.

Wikipedia Article graph Wikipedia articles are heavily linked, as links can be easily inserted while editing an article. If we treat each article as a node, and each link between articles as an edge running from one node to another, then Wikipedia articles form a directed graph (see right side of Figure 4.1). The article graph has been targeted by numerous studies, and is not addressed in this thesis. Buriol et al. (2006) analyze the development of the article graph over time, and find that some regions are fairly stable, while others are advancing quickly. Zlatic et al. (2006) give a comprehensive overview of the graph parameters for the largest languages in Wikipedia. Capocci et al. (2006) study the growth of the article graph and show that it is based on preferential attachment (Barabasi and Albert, 1999). Voss (2005) shows that the article graph is scale-free and grows exponentially.

Wikipedia Category graph In Wikipedia, each article can link to an arbitrary number of categories, where each category is a kind of semantic tag for that article.

A category links to all articles in this category. Thus, article graph and WCG are heavily interlinked (see Figure 4.1), and most studies (Capocci et al., 2006; Zlatic et al., 2006) have not treated them separately. However, categories in Wikipedia are organized in a taxonomy-like structure (see left side of Figure 4.1 and Figure 4.4-a), while relations between articles cannot be easily classified. Each category can have an arbitrary number of subcategories, where a subcategory is typically established because of a HYPONYMY or MERONYMY relation. For example, a category *vehicle* has subcategories like *aircraft* or *watercraft*. Thus, the WCG is very similar to semantic wordnets like WordNet or GermaNet. As Wikipedia does not strictly enforce a taxonomic category structure, cycles and disconnected categories are possible, but rare. In the snapshot of the German Wikipedia[1] from May 15, 2006, the largest connected component in the WCG contains 99,8% of all category nodes, as well as 7 cycles. For our analysis, we only consider this largest connected component.

Wiktionary Graph Wiktionary entries are linked through explicitly encoded relations. As the English language edition does not contain HYPERNYMY or HYPONYMY relations, we limit our study to the German language edition from Oct 9, 2007. We are not aware of any studies on the graph-theoretic properties of Wiktionary. The analysis of the Wiktionary graph as described in this thesis is based on joint work with Konstantina Garoufi (Garoufi et al., 2008a).

Analysis For our analysis, we treat the directed graphs in GermaNet, the WCG and the German Wiktionary as an undirected **graph** $G = (V, E)$, because the relations connecting categories are reversible. V is a set of vertices or nodes. E is a set of unordered pairs of distinct vertices, called edges. Each page is treated as a **node** n, and each link between pages is modeled as an **edge** e running between two nodes.[2]

Following Steyvers and Tenenbaum (2005), we characterize the graph structure of a lexical-semantic resource in terms of a set of graph parameters: The **degree** k of a node is the number of edges that are connected with this node. Averaging over all nodes gives the **average degree** $\overline{k}$. The degree distribution $P(k)$ is the probability that a random node will have degree k. In some graphs (like the WWW), the degree distribution follows a power law $P(k) \approx k^{-\gamma}$ (Barabasi and Albert, 1999). We use the **power law exponent** γ as a graph parameter.

A **path** $p_{i,j}$ is a sequence of edges that connects a node n_i with a node n_j. The **path length** $l(p_{i,j})$ is the number of edges along that path. There can be more than one path between two nodes. The **shortest path length** L is the minimum of all these paths, i.e. $L_{i,j} = \min l(p_{i,j})$. Averaging over all nodes gives the **average shortest path length** $\overline{L}$. The **diameter** D is the maximum of the shortest path lengths between all pairs of nodes in the graph.

The **cluster coefficient** of a certain node n_i can be computed as

$$C_i = \frac{T_i}{\frac{k_i(k_i-1)}{2}} = \frac{2T_i}{k_i(k_i-1)}$$

[1] Wikipedia can be downloaded from `http://download.wikimedia.org/`

[2] Newman (2003) gives a comprehensive overview of the theoretical aspects of graphs.

where T_i refers to the number of edges between the neighbors of node n_i and $k_i(k_i - 1)/2$ is the maximum number of edges that can exist between the k_i neighbors of node n_i.[3] The cluster coefficient C for the whole graph is the average of all C_i. In a fully connected graph, the cluster coefficient is 1.

Table 4.1 shows our results on the WCG as well as the corresponding values for other well-known graphs and lexical-semantic networks. We compare our empirically obtained values with the values expected for a random graph. Following Zlatic et al. (2006), the cluster coefficient C for a random graph is

$$C_{random} = \frac{(\overline{k}^2 - \overline{k})^2}{|V|\overline{k}}$$

The average path length (Watts and Strogatz, 1998) for a random network can be approximated as:

$$\overline{L}_{random} \approx \log|V| \;/\; \log\overline{k}$$

From the analysis, we conclude that all graphs in Table 4.1 are small world graphs (see Figure 4.4-c). Small world graphs (Watts and Strogatz, 1998) contain local clusters that are connected by some long range links leading to low values of $\overline{L}$ and D. Thus, small world graphs are characterized by (i) small values of $\overline{L}$ (typically $\overline{L} \gtrsim \overline{L}_{random}$), together with (ii) large values of C ($C \gg C_{random}$).

Additionally, all semantic networks are scale-free graphs, as their degree distribution follows a power law. Structural commonalities between the graphs in Table 4.1 are assumed to result from the growing process based on preferential attachment (Capocci et al., 2006).

Our analysis shows that WordNet, GermaNet, the Wikipedia category graph, and Wiktionary are (i) scale-free, small world graphs, and (ii) have a very similar parameter set. Thus, we conclude that algorithms designed to work on the graph structure of WordNet can be adapted to the WCG and the German Wiktionary.

4.2 Gloss Based Measures

Gloss based measures rely on word overlaps between concept definitions. Dictionaries, some wordnets, and Wiktionary usually contain such definitions. Gloss based measures can then be directly applied. Thesauri or wordnets lacking glosses can use pseudo glosses (Gurevych, 2005) as a substitute.

Adapting to GermaNet In the case of GermaNet, that contains only few glosses, we need to construct pseudo glosses (as described in Section 3.2) as a proxy for textual descriptions of a concept.

Adapting to Wikipedia Each Wikipedia article contains a rather long textual description that can be used for gloss based measures, but also contains a lot of unrelated terms. Thus, the first paragraph of an article can be used as a more concise textual description.

[3] In a social network, the cluster coefficient measures how many of my friends (neighboring nodes) are friends themselves.

Parameter	**Actor**	**Power**	**C.elegans**	**AN**	**Roget**	**WordNet**	**GermaNet**	**Wiki$_{\mathbf{Art}}$**	**WCG**	**Wiktionary**
$\|V\|$	225,226	4,941	282	5,018	9,381	122,005	42,129	190,099	27,865	20,011
D	-	-	-	5	10	27	25	-	17	17
$\overline{k}$	61.0	2.67	14.0	22.0	49.6	4.0	3.8	-	3.54	5.80
γ	-	-	-	3.01	3.19	3.11	1.96	2.45	2.12	2.28
$\overline{L}$	3.65	18.7	2.65	3.04	5.60	10.56	8.77	3.34	7.18	5.03
$\overline{L}_{random}$	2.99	12.40	2.25	3.03	5.43	10.61	7.65	$\sim$ 3.30	$\sim$ 8.10	5.31
C	0.79	0.08	0.28	0.186	0.87	0.027	0.016	$\sim$ 0.04	0.012	0.082
C_{random}	0.0003	0.005	0.05	0.004	0.613	0.0001	0.0001	$\sim$ 0.006	0.0008	0.0005

Table 4.1: Parameter values for different graphs.

Values for *Actor* (collaboration graph of actors in feature films), *Power* (the electrical power grid of the western United States) and *C.elegans* (the neural network of the nematode worm C. elegans) are from (Watts and Strogatz, 1998). Values for AN (a network of word associations by Nelson et al. (1998)), Roget's thesaurus and WordNet are from (Steyvers and Tenenbaum, 2005). Values for $Wiki_{art}$ (German Wikipedia article graph) are from (Zlatic et al., 2006). We took the values for the page set labelled M on their website containing 190,099 pages for German, as it comes closest to a graph of only articles. Values marked with '-' in the table were not reported in the studies. We computed the values for GermaNet 5.0, the Wikipedia category graph (WCG), and the German Wiktionary (Garoufi et al., 2008a).

n = number of nodes; $\overline{k}$ = average node degree; γ = power law exponent of the degree distribution; $\overline{L}$ = average shortest path length; D = diameter; C = cluster coefficient; Parameters with a "random" subscript are expected values for a random graph of the same size and density.

4.3 Concept Vector Based Measures

In the original definition by Gabrilovich and Markovitch (2007), the concept vector based measure relies on textual representations of concepts derived from Wikipedia articles. For the first time, we generalize this measure to each semantic resource where we can retrieve or construct a textual description for each concept (Zesch et al., 2008b).

Adapting to WordNet WordNet contains glosses that can be seen as very short 'articles' describing the concepts expressed by WordNet synsets. For example, the term *car* is contained in more than 250 glosses of WordNet concepts including nouns (e.g. *polishing*, "every Sunday he gave his car a good polishing"), verbs (e.g. *damage*, "she damaged the car when she hit the tree"), and adjectives (e.g. *unfastened*, "the car door was unfastened"). Each of these concepts leads to a non-zero entry in the resulting concept vector for *car*. When computing semantic relatedness, the whole vector is taken into account, and in that way all the implicit relations to *car*-related concepts are encoded in the glosses.

Adapting to GermaNet We use pseudo glosses as described in Section 3.2.

Adapting to Wiktionary Wiktionary combines a collaborative construction approach with explicitly encoded lexical-semantic relations, glosses, translations etc. Thus, it provides a rich set of focused additional knowledge associated with each concept. We can create a rich textual description of a concept by using its gloss in combination with a pseudo gloss created from all other relation types.

4.4 Chapter Summary

In this chapter, we showed how state-of-the-art semantic relatedness measures defined on a specific semantic resource can be adapted to most other semantic resources. We described in detail the problematic cases of adapting path based and information content based measures to Wikipedia. We verified the validity of this adaptation process by a graph-theoretic analysis showing that all semantic resource graph structures have similar properties. We then described how we generalized gloss based and vector based measures to any semantic resource containing a textual description of the included concepts.

Chapter 5

Evaluating Semantic Relatedness Measures

The prevalent approaches for evaluating semantic relatedness measures are (i) mathematical analysis, (ii) application-specific evaluation, (iii) correlating semantic relatedness with human judgments, and (iv) solving word choice problems.

Mathematical analysis (Lin, 1998) can assess a measure with respect to some formal properties, e.g. whether it is a metric,[1] but cannot tell us whether a measure closely resembles human judgments, or how it performs in a certain application.

The latter question is tackled by *application-specific evaluation*, where a measure is tested within the framework of a usually complex application, e.g. word sense disambiguation (Patwardhan et al., 2003) or malapropism detection (Budanitsky and Hirst, 2006). However, application specific evaluation entails influence of parameters besides the measure of semantic relatedness being tested. Gurevych and Strube (2004) evaluated a set of WordNet-based semantic similarity measures for the tasks of dialog summarization, and did not find any significant differences in their performance. Rather, the performance of a specific measure is tightly correlated with the properties of the underlying semantic resource, as shown by Gurevych et al. (2007) when evaluating semantic relatedness measures in an information retrieval task. Budanitsky and Hirst (2006) evaluated semantic relatedness measures on the task of real word error detection and found that the choice of a specific measure influences detection performance.

The remaining approaches, *comparison with human judgments* (described in Section 5.1) and *solving word choice problems* (described in Section 5.2), are used in this thesis to gain deeper insights into the nature of semantic relatedness, as well as the performance of measure types and their dependence on a semantic resource.

5.1 Comparison with Human Judgments

Semantic relatedness measures can be evaluated using two different correlation methods. The first method is to correlate the scores computed by a semantic relatedness measure with the judgments provided by humans. For example, on a 0–4 scale,

[1] A metric fulfills the mathematical criteria: (i) $dist(c_1, c_2) \geq 0$; (ii) $dist(c_1, c_2) = 0 \Leftrightarrow c_1 = c_2$; (iii) $dist(c_1, c_2) = dist(c_2, c_1)$; and (iv) $dist(c_1, c_3) \leq dist(c_1, c_2) + dist(c_2, c_3)$.

where 4 is maximum relatedness, the pair (*car* – *drive*) might get a human judgment of 3.9 and a score of 3.7 from a measure. A second pair (*car* – *eat*) only gets 1.1 (human) and 0.4 (machine).

The Pearson product-moment correlation coefficient r can be employed as an evaluation measure. It indicates how well the results of a measure resemble human judgments, where a value of 0 means no correlation and a value of 1 means perfect correlation. Pearson's r is calculated as:

$$r = \frac{n\left(\sum x_i y_i\right) - \left(\sum x_i\right)\left(\sum y_i\right)}{\sqrt{n\left(\sum x_i^2\right)\left(\sum x_i\right)^2}\sqrt{n\left(\sum y_i^2\right)\left(\sum y_i\right)^2}} \tag{5.1}$$

where x_i is the i-th element in the list of human judgments and y_i is the corresponding i-th element in the list of semantic relatedness values computed by a certain measure.

The second method is correlating word pair rankings. In a ranking task, a human and a measure would simply rank the pair (*car* – *automobile*) higher than (*car* – *garden*). The ranking produced on the basis of the measure is compared to the one produced on the basis of human judgments. The quality of such a ranking is quantified by the Spearman rank order correlation coefficient ρ, where a value of 0 means no correlation and a value of 1 means perfect correlation. As a semantic relatedness measure normally outputs a numerical value within the range $[0, 1]$ instead of ranks, the raw values are converted into ranks. Then, d_i is the difference between the ranks of x_i and y_i. If there are no tied ranks, Spearman's ρ can be calculated with the simplified formula:

$$\rho = 1 - \frac{6\sum d_i^2}{n(n^2-1)}$$

In case of tied ranks, Spearman's ρ is calculated as the Pearson correlation of ranks using Formula (5.1).

Existing work on computing semantic relatedness often employed Pearson correlation as an evaluation measure. However, this suffers from some limitations:

- Pearson correlation is very sensitive to outliers. Even a single outlier might yield fundamentally different results. This is visualized by Anscombe's quartet (Anscombe, 1973), a set of four scatterplots showing relationships with exactly the same mean, standard deviation, regression line, and Pearson correlation of $r = 0.81$ (see Figure 5.1). In the lower figures, a single outlier is sufficient to disturb a perfect correlation (bottom left) or produce a high correlation in a fully non-linear relationship (bottom right).[2]

- Pearson correlation measures the strength of the *linear* relationship between human judgment and semantic relatedness scores computed by a measure. If a

[2]A good real-life example gives the Les86 measure that showed a remarkable difference between a non-significant Pearson correlation r and a very high Spearman rank correlation ρ in our initial re-implementation of the measure. If both words in a pair are mapped to the same concept, e.g. (*car* – *automobile*), they have identical glosses resulting in an exceptionally high overlap value that represents an outlier in the dataset. This led to a very low, non significant Pearson correlation coefficient. If we smooth the distribution by using the natural logarithm of the relatedness values returned by Les86, the resulting Pearson correlation coefficient increases to the level of the Spearman rank correlation.

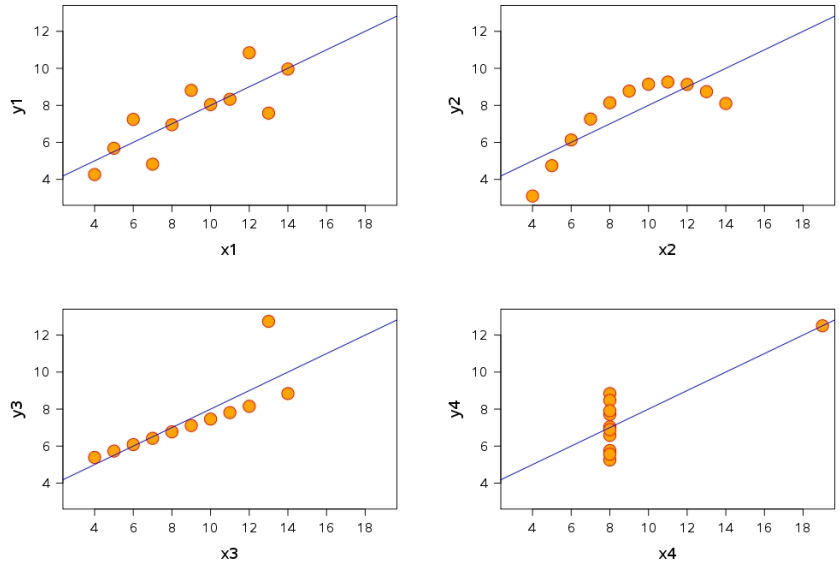


Figure 5.1: 'Anscombe's quartet' showing relationships with exactly the same mean, standard deviation, regression line, and Pearson correlation of $r = 0.81$. (Adapted from `http://en.wikipedia.org/wiki/Image:Anscombe.svg`)

relationship is not linear, results are flawed. For example, the upper right chart in Figure 5.1 shows a non-linear relation that cannot be correctly measured using Pearson correlation.

- Pearson correlation requires the two random variables (the vectors) to be normally distributed and measured on interval scales. In Figure 5.1, only the variables in the upper left plot fulfill the prerequisite of being normally distributed. However, the real distribution of relatedness values is largely unknown. The values of most samples (small subsets of word pairs judged by humans) are not normally distributed. Recent findings (Budanitsky and Hirst, 2006; Zesch and Gurevych, 2006) even indicate that the relatedness values as perceived by humans are not interval scaled.

In contrast to these limitations, Spearman's rank correlation coefficient is robust against outliers, and can also measure the strength of non-linear relationships. It does not pose the assumption of interval scales, i.e. it can be used for variables measured on the ordinal level. Additionally, Spearman's ρ does not make any assumptions about the distribution of the vectors being compared. However, using Spearman rank correlation also has some disadvantages: From the statistical literature, it is known that Spearman's ρ tends to give higher values than Pearson's r for datasets with many tied ranks. For some applications that rely on thresholding semantic relatedness scores, a measure that yields a perfect ranking might be of little use, if the differences between semantic relatedness scores are too small to be sensibly thresholded.

For comparison with previous results, we report both Pearson's r and Spearman's ρ. Still, we recommend using Spearman rank correlation in future experiments, as this evaluation is more objective, if the performance of semantic relatedness measures

Dataset	Year	Language	# Pairs	PoS	Scores	# Subjects
RG–65	1965	English	65	N	$[0, 4]$	51
MC–30	1991	English	30	N	$[0, 4]$	38
Res–30	1995	English	30	N	$[0, 4]$	10
Fin–353	2002	English	353	N, V, A	$[0, 10]$	13/16
Fin1–153			153			13
Fin2–200			200			16
YP–130	2006	English	130	V	{0,1,2,3,4}	6
Gur–65	2005	German	65	N	{0,1,2,3,4}	24
Gur–30	2005	German	30	N	{0,1,2,3,4}	24
Gur–350	2006	German	350	N, V, A	{0,1,2,3,4}	8
ZG–222	2006	German	222	N, V, A	{0,1,2,3,4}	21

Table 5.1: Evaluation datasets for comparison with human judgments.

has to be evaluated intrinsically. As Pearson correlation and Spearman correlation are not directly comparable and might yield very different results under certain conditions, special care must be taken when comparing and interpreting such results.

5.1.1 Datasets

Evaluation datasets for correlation analysis are created by asking human annotators to judge the relatedness of presented word pairs. The gold standard score assigned to a word pair is the average score over all human judges. For evaluation, a gold standard dataset is then correlated with the semantic relatedness scores computed by a particular measure.

An upper bound for the performance of a measure on a dataset is the inter-annotator agreement (**InterAA**), i.e. the amount of mutual agreement between human judges. InterAA is computed as the average pairwise Pearson correlation between human judges. As the distribution of Pearson's r is left-skewed, we cannot simply average the correlations, but have to use a Fisher Z-value transformation.

$$Z = \frac{\ln(1 + r) - \ln(1 - r)}{2} \tag{5.2}$$

The Fisher Z-values can then be averaged and transformed back to a Pearson's r value to get the average Pearson correlation. In contrast to InterAA, the *intra*-annotator agreement (**IntraAA**) measures the agreement of a judge with herself over time. It is computed analogously to the inter-annotator agreement. Unfortunately, only few experiments with intra-annotator agreement have been performed in previous work.

Several evaluation datasets have been created so far. Table 5.1 gives an overview of the datasets, while Table 5.2 shows the InterAA and IntraAA values. In the seminal work by Rubenstein and Goodenough (1965), similarity judgments were obtained from 51 test subjects on 65 noun pairs written on paper cards. Test subjects were instructed to order the cards according to the "similarity of meaning" and then assign a continuous similarity value $[0, 4]$ to each card. The final dataset contains 65 English noun pairs and will be referenced as **RG–65** in the remainder of this thesis. No InterAA was reported for this dataset, but Pirró and Seco (2008) repeated the experiment yielding an InterAA for native speakers of $r = .80$. Miller and Charles

		Correlation r	
Dataset	Language	InterAA	IntraAA
RG–65	English	(.80)	.85
MC–30	English	(.90)	-
Fin–353	English	-	-
Fin1–153	English	.73	-
Fin2–200	English	.55	-
YP–130	English	.87	-
Gur–65	German	.81	-
Gur–30	German	-	-
Gur–350	German	.69	-
ZG–222	German	.49	.65

Table 5.2: Inter- and intra-annotator agreement on evaluation datasets. Missing values are not available from the references.

(1991) (**MC–30**) replicated the experiment with 38 test subjects judging on a subset of 30 pairs taken from the original 65 pairs. This experiment was again replicated by Resnik (1995) with 10 subjects yielding an InterAA of $r = .90$.

As creating datasets of this kind is time-consuming and costly, most work on evaluating semantic relatedness measures focused on such small scale experiments restricted to nouns (Li et al., 2003; Budanitsky and Hirst, 2006; Patwardhan and Pedersen, 2006). This leads to overfitting of algorithms to these specific datasets and the employed semantic resource. Many algorithms yield near human performance on these particular datasets using WordNet as a semantic resource. This is due to the strongly related word pairs in these datasets being only related by classical lexical-semantic relations that are well modelled in WordNet.

We argue that previous evaluations restricted to those datasets were limited with respect to (i) the number of word pairs involved, (ii) the parts-of-speech of word pairs, (iii) approaches to select word pairs (manual vs. automatic, analytic vs. corpus based), and (iv) the kinds of semantic relations that hold between word pairs. However, an evaluation involving the aspects described above is crucial to understand the properties of a specific measure and the results obtained under certain experimental conditions (e.g. the semantic resource used). First of all, a comprehensive evaluation of semantic relatedness measures requires a higher number of word pairs. However, the original experimental setup is not scalable as ordering several hundred paper cards is a cumbersome task. Furthermore, semantic relatedness is an intuitive concept and being forced to assign fine-grained continuous values is felt to overstrain the test subjects.

Finkelstein et al. (2002) created a larger dataset for English containing 353 word pairs (**Fin–353**) including also the 30 word pairs from MC30. This dataset has been criticized for being culturally biased (Jarmasz and Szpakowicz, 2003). Another problem with this dataset is that it consists of two subsets, which have been annotated by different human judges. We performed further analysis of their dataset and found that the InterAA differs considerably for the two subsets ($r = 0.73$ vs. $r = 0.55$). Therefore, we treat them as independent datasets **Fin1–153** and **Fin2–200** henceforth.

Yang and Powers (2006) created a dataset (**YP–130**) that contains 130 verb pairs. They report a high InterAA of $r = .87$. As this dataset contains only

verbs, the evaluation will be particularly informative about the ability of a semantic relatedness measure to estimate verb relatedness.

Several German datasets have also been created (see Table 5.1). Gurevych (2005) conducted experiments with a German translation of the English RG65 dataset (abbreviated as **Gur–65**). The subset of the Gur–65 dataset with the translated word pairs corresponding to the MC30 dataset is called **Gur–30**. As both the Gur–65 and the Gur–30 dataset are small and contain only noun pairs connected by either SYNONYMY or HYPONYMY, she conducted a follow-up study and collected a larger dataset containing 350 word pairs (**Gur–350**). It contains nouns, verbs and adjectives that are connected by classical and non-classical relations (Morris and Hirst, 2004). However, word pairs for this dataset are biased towards strong classical relations, as they were manually selected. For that reason, Zesch and Gurevych (2006) propose an approach to create word pairs from domain specific corpora using a semi-automatic process (see Section A.2 for a more detailed description). The resulting **ZG–222** dataset contains 222 domain specific word pairs that are connected by different kinds of lexical-semantic relations. As human judgments on domain specific word pairs depend on the domain knowledge of the judges, the InterAA is relatively low ($r = 0.49$). For example, the word pair (*Extruder – Gummi*)[3] had an extremely large variance of human judgment. Subjects which were aware that extruders are used to shape rubber assigned a very high score, while subjects being not aware of that fact assigned a very low score. Thus, creating domain-specific datasets requires subjects that are domain experts. Due to the low InterAA, we do not use this dataset in our evaluation.

In psycholinguistics, relatedness of words can also be determined through association tests (Schulte im Walde and Melinger, 2005). Subjects are presented a term (e.g. *lemon*) and their spontaneous responses are recorded (e.g. *lime*, *sour*, *squeeze* etc.). Results of such experiments are hard to quantify and cannot easily serve as the basis for evaluating semantic relatedness measures.

In Chapter 6, we describe the results of evaluating semantic relatedness on the English and German datasets presented in this section. We now turn to the second evaluation task: solving word choice problems.

5.2 Solving word choice problems

This approach to the evaluation of semantic relatedness measures relies on word choice problems (Jarmasz and Szpakowicz, 2003; Turney, 2006). A word choice problem consists of a target word and four candidate words or phrases. The objective is to pick the one that is most closely related to the target. An example problem is given below. There is always only one correct candidate, 'a)' in this case.

beret

a) round cap	b) cap with horizontal peak
c) wedge cap	d) helmet

The relatedness between the target 'beret' and each of the candidates is computed by a semantic relatedness measure, and the candidate with the maximum semantic

[3]English: (*extruder – rubber*)

relatedness value is chosen. We lemmatize the target and all candidates. This is especially beneficial for German words that can be highly inflected.

If two or more candidates are equally related to the target, then the candidates are said to be tied. If one of the tied candidates is the correct answer, then the problem is counted as correctly solved, but the corresponding score is reduced. We assign a score s_i of $\frac{1}{\#\text{ of tied candidates}}$ (in effect approximating the score obtained by randomly guessing one of the tied candidates). Thus, a correctly solved problem without ties is assigned a score of 1.

If a phrase or a multiword expression is used as a candidate and cannot be found in the semantic resource, we remove stopwords (prepositions, articles, etc.) and split the candidate phrase into component words. For example, the target *beret* in the above example has *cap with horizontal peak* as one of its answer candidates. The candidate phrase is split into its component content words *cap*, *horizontal*, and *peak*. We compute semantic relatedness between the target and each phrasal component and select the maximum value as the relatedness between the target and the candidate. If the target or all candidates cannot be found in the semantic resource, a semantic relatedness measure does not attempt to solve the problem. The overall score S of a semantic relatedness measure is the sum of the scores yielded on the single problems $S = \sum_{wp_i \in A} s(wp_i)$, where A is the set of word choice problems that were attempted by the measure, and wp_i is a certain word choice problem.

Jarmasz and Szpakowicz (2003) use the overall score S for evaluation. However, this evaluation approach is problematic, as a measure that attempts more problems may get a higher score just from random guessing. Mohammad et al. (2007) used precision, recall and F-measure for evaluation. For word choice problems, recall is defined as $R = \frac{S}{n}$, where n is the total number of word choice problems. Under that definition, if the score S is 0 then a semantic relatedness measure has a recall of 0, regardless of how many word choice problems were attempted. This stands in contrast to the use of precision and recall in information retrieval, where just retrieving all documents will always give a recall of 1, regardless whether they are relevant or not. For word choice problems, just attempting all problems will only yield a recall of 1 if all attempted problems are correctly solved at the same time. Thus, for this task, recall is of very limited value for judging about the performance of a semantic relatedness measure. We therefore decided not to use precision and recall, but evaluate the word choice problems using accuracy and coverage instead. We define accuracy as

$$Acc = \frac{S}{|A|}$$

where S is the overall score as defined above and A is the number of word choice problems that were attempted by the semantic relatedness measure. Coverage is then defined as

$$Cov = \frac{|A|}{n}$$

where n is the total number of word choice problems. Accuracy indicates how many of the attempted problems could be answered correctly, and coverage indicates how many problems were attempted.

The overall performance of a measure needs to take accuracy *and* coverage into account, as a measure might get a better coverage by sacrificing accuracy and vice

versa. Thus, we define the combined evaluation metric H as

$$H = \frac{2 \cdot Acc \cdot Cov}{Acc + Cov}$$

i.e. the harmonic mean of accuracy and coverage. This is in analogy to the F_1-measure in information retrieval which is the harmonic mean of precision and recall.

5.2.1 Datasets

The English dataset contains 300 word choice problems collected by Jarmasz and Szpakowicz (2003). We collected a German dataset from the January 2001 to December 2005 issues of the German-language edition of Reader's Digest (Wallace and Wallace, 2005). We discarded 44 problems that had more than one correct candidate, and 20 problems that used a phrase instead of a single term as the target. The remaining 1008 problems form our German word choice dataset, which is significantly larger than any of the previous datasets employed in this type of evaluation.

We tested human performance on a subset of 200 manually selected German word choice problems using 41 native speakers of German. Human coverage on word choice problems is always perfect, as the experimental setting did not allow to skip word choice problems. We found that human accuracy on this task strongly depends on the level of language competence of the subjects. Average accuracy was $Acc = .71$ with $\sigma = .10$. We also observed a Pearson correlation $r = .69$ between a subject's accuracy and her age (statistically significant, two tailed t-test with $\alpha = .01$). The highest accuracy was .91 (by the oldest subject), the lowest .45 (by the youngest subject).

5.3 Chapter Summary

In this chapter, we described two intrinsic evaluation approaches that are used in this thesis: *comparison with human judgments* and *solving word choice problems*. We described our evaluation setup and the available datasets. In the next chapter, we report the results obtained using this setup.

Chapter 6

Experiments and Results

In this chapter, we first describe in Section 6.1 the configuration of semantic relatedness measures used in our experiments. In Section 6.2, we then present the results of the first evaluation task *comparison with human judgments* and in Section 6.3 of the second task *solving word choice problems*. We finally summarize our findings in Section 6.4.

6.1 Configuration of Measures

We implement the semantic relatedness measures using the interoperability framework presented in Section 2.4. This ensures that all resources use exactly the same implementation of a certain semantic relatedness measure, making the results better comparable between resources. In previous work, we did not use the interoperability framework, but applied the measures as available in software packages (e.g. Perl WordNet::Similarity package (Patwardhan and Pedersen, 2006) or re-implemented them using native APIs (e.g. GermaNet-API[1], JWPL or JWKTL (Zesch et al., 2008a)). Differences of the results reported in this thesis to previously reported results are due to these changes.

WordNet We use WordNet 3 and integrate it into the interoperability framework using the JWNL WordNet API.[2] In the framework, we treat each WordNet synset as a single concept. A concept's textual representation for constructing the concept vectors can either be its gloss (together with the example sentences), or a pseudo gloss. We construct pseudo glosses by concatenating the lemmas of all concepts that are reachable within a radius of three concept nodes from the original concept.[3] In the following, we indicate the use of pseudo glosses with the suffix '-pseudo'.

GermaNet We use GermaNet 5.0 and integrate it into the interoperability framework using the native GermaNet-API. We construct pseudo glosses by concatenating the lemmas of all concepts that are reachable within a radius of three concept nodes

[1] `http://projects.villa-bosch.de/nlpsoft/gn_api/index.html`
[2] `http://jwordnet.sourceforge.net/`
[3] Optimized configuration as reported by Gurevych (2005).

from the original concept. We use the same pseudo glosses as textual representations for constructing concept vectors (we indicate the use of pseudo glosses with the suffix '-pseudo').

Wikipedia We use the JWPL Wikipedia API (see Appendix A.1) to integrate the English and German Wikipedia dumps from February 6, 2007 into the interoperability framework. Path based and information content based semantic relatedness measures that were originally defined on WordNet are adapted to Wikipedia as described in Chapter 4.

For gloss based and vector based measures, we differentiate between considering the full Wikipedia article as the textual representation, or just the first paragraph. The first paragraph usually contains a definition of the concept described in that article. As some words in the latter parts of an article are likely to describe less important or even contradicting topics, we expect this refined measures to yield a better accuracy by trading in some coverage. In the following, we flag the measures that only use the first paragraph with the suffix '-first'.

When using the full English Wikipedia to construct concept vectors, we prune the concept space for performance reasons by only considering articles as concepts if they contain at least 100 words and have more than 5 inlinks and 5 outlinks.[4] When using only the first paragraph of each article, we do not need any performance tuning and consider all articles as concepts.

Wiktionary We use the JWKTL Wiktionary API (Zesch et al., 2008a) to integrate the English Wiktionary dump from October 16, 2007 and the German Wiktionary dump from October 9, 2007 into the interoperability framework. We do not report path based or IC based results here, as the English Wiktionary contains only very few Hypernymy and Hyponymy relations, limiting the applicability of most path based and IC based measures. Normal gloss based measures only use the short glosses from a Wiktionary entry, while pseudo glosses are constructed like for the other resources by concatenating the lemmas of all concepts reachable within a radius of three concept nodes from the original concept. As the number of relations in Wiktionary is quite high, we only consider Antonymy, Holonymy, Hypernymy, Hyponymy, Meronymy, See also, and Synonymy in order to stay as comparable as possible with the other resources. Textual representations for the vector based measures are created by concatenating the contents of all relation types offered by JWKTL for each Wiktionary entry. From the wide range of lexical-semantic relations in Wiktionary, we only use Antonymy, Categories, Characteristic word combinations, Coordinate terms, Derived terms, Etymology, Examples, Glosses, Holonymy, Hypernymy, Hyponymy, Meronymy, See also, Synonymy, and Troponymy.

6.2 Comparison with Human Judgments

In this section, we report the results obtained when correlating human judgments on semantic relatedness of word pairs with the values computed by semantic relat-

[4]Same configuration as used by Gabrilovich and Markovitch (2007).

(a) Results on English datasets.

Dataset			MC–30	RG–65	Fin1–153	Fin2–200	YP–130
Word pairs used			30	65	144	190	80
		Type	ρ	ρ	ρ	ρ	ρ
WordNet	Rad89	PL	.84	**.84**	.41	.21	.75
	WuP94	PL	.80	.79	.47	.21	.72
	LC98	PL	.84	**.84**	.41	.21	.75
	Res95	IC	.79	.81	**.49**	.18	.76
	JC97	IC	**.88**	**.84**	.48	.18	.75
	Lin98	IC	.78	.82	**.49**	.19	**.77**
Wikipedia	Rad89	PL	*.33*	.36	.35	*.20*	*.09*
	WuP94	PL	*.38*	.37	.38	*.19*	*.07*
	LC98	PL	*.33*	.36	.35	*.20*	*.09*
	Res95	IC	.54	.43	.28	*.20*	*.18*
	JC97	IC	*.15*	*.13*	*.07*	*.06*	*.05*
	Lin98	IC	.55	.45	*.24*	**.22**	*.20*

(b) Results on German datasets.

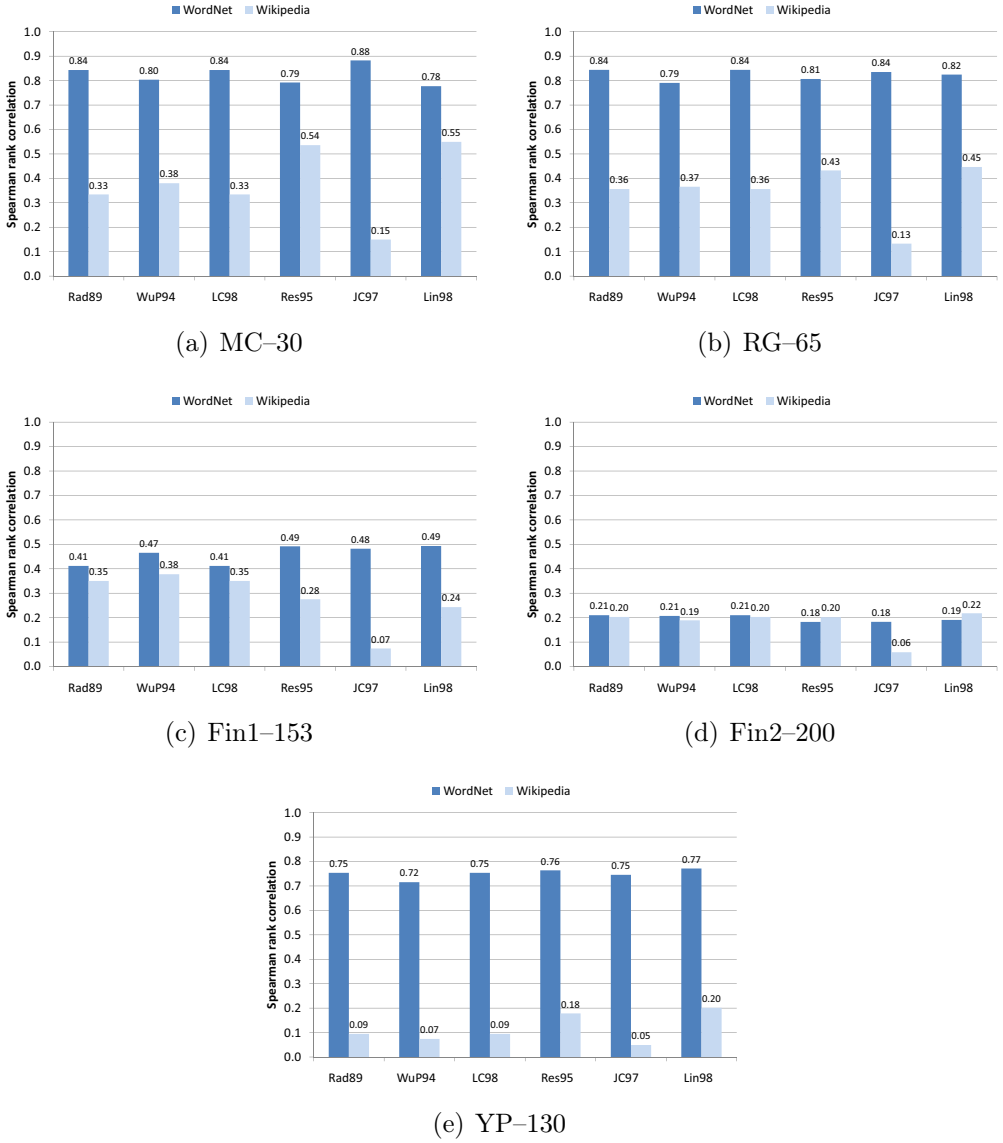
Dataset			Gur–30	Gur–65	Gur–350
Word pairs used			27	53	101
		Type	ρ	ρ	ρ
GermaNet	Rad89	PL	**.68**	**.69**	**.43**
	WuP94	PL	.40	.49	*.31*
	LC98	PL	**.68**	**.69**	**.43**
	Res95	IC	.56	.54	.37
	JC97	IC	.61	.50	*.23*
	Lin98	IC	.56	.54	.37
Wikipedia	Rad89	PL	.57	.38	.39
	WuP94	PL	.62	.37	.38
	LC98	PL	.57	.38	.39
	Res95	IC	.63	.41	.42
	JC97	IC	.59	*.32*	*.36*
	Lin98	IC	.63	.40	.41

Table 6.1: Spearman correlation of path based (PL) and information content based (IC) measures with human judgments on English and German datasets. Best values for each dataset are in bold. Non-significant correlations are in italics (two tailed t-test, $\alpha = .05$).

edness measures. We first have a detailed look on each type of semantic relatedness measures, and then compare the performance of measure types as well as semantic resources. We finally have a look at the coverage of semantic resources and at the influence of the growth of semantic resources on the results.

6.2.1 Adapted Path and IC Based Measures

Tables 6.1 (a) and (b) give an overview of the results of path and IC based measures on the English and German datasets. To ensure a fair comparison of the measures' performance, we only use the subset of word pairs that is covered by all measures as indicated in the tables. We only present results in terms of Spearman rank



(a) MC–30 (b) RG–65

(c) Fin1–153 (d) Fin2–200

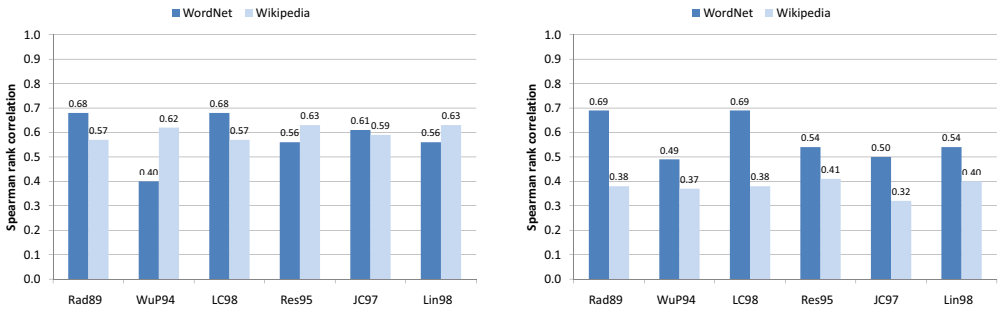
(e) YP–130

Figure 6.1: Comparison of individual measures applied to WordNet and Wikipedia.

correlation in this section. The complete results containing also Pearson correlation values for comparison with previous work can be found in Appendix B.

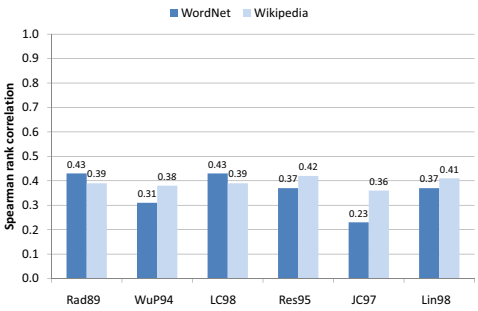
When looking at the results, we find that the Spearman correlation values of the two measures Rad89 $dist_{\text{Rad89}} = l(c_1, c_2)$ and LC98 $rel_{\text{LC98}} = -\log \frac{l(c_1,c_2)+1}{2 \cdot depth}$ are always equal, as the denominator $2 \cdot depth$ in the LC98 formula is a constant and does not change the rank of a word pair, the logarithm does neither. Thus, LC98 is just a variant of Rad89 that scales the obtained path lengths to be better linearly correlated with human judgments.

Figures 6.1 and 6.2 visualize the performance of a certain measure using WordNet and GermaNet as compared with its adaptation to Wikipedia. We first focus on an analysis of the English results. Figures 6.1(a) and 6.1(b) give the results for the datasets containing only noun pairs connected by classical lexical-semantic relations. All measures work better using WordNet as compared to using Wikipedia. This is also true for the special case of the verb similarity dataset (YP–130) shown in



(a) Gur–30

(b) Gur–65



(c) Gur–350

Figure 6.2: Comparison of individual measures applied to GermaNet and Wikipedia.

Figure 6.1(e). However, the differences are even more pronounced in this case, because verb similarity is well modelled in WordNet, while Wikipedia focuses on nouns and named entities. Figures 6.1(c) and 6.1(d) show a different picture for the two datasets containing also cross part-of-speech pairs and word pairs connected by non-classical relations. The performance level is generally much lower than on the other datasets, and the difference between a measure using WordNet or being adapted to Wikipedia is small.

In contrast to the previous findings (Strube and Ponzetto, 2006), we do not find performance increases when adapting path based measures from WordNet to Wikipedia. However, we used a different version of Wikipedia, a different disambiguation strategy, and conducted experiments with the two subsets of the Fin–353 dataset instead of the full dataset. Thus, the results may not be directly comparable.

For the German datasets in Figure 6.2, we do not get such a clear picture as on the English datasets. Some measures work better using GermaNet, while some of them work better using Wikipedia. However, the measures adapted to Wikipedia never yield significant performance gains over their counterparts working on GermaNet. Thus, the findings on the English and German datasets show that the adaptation of WordNet-defined path and IC based measures to Wikipedia has been successful on a formal level, but the obtained results do not justify the efforts given the higher computational costs of using the larger Wikipedia as compared to WordNet or GermaNet.

(a) Results on English datasets.

Dataset		MC–30	RG–65	Fin1–153	Fin2–200	YP–130
Word pairs used		30	65	146	193	90
		ρ	ρ	ρ	ρ	ρ
WordNet	Les86	.43	.53	*.20*	*.01*	.54
WordNet-pseudo	Gur05	**.82**	**.78**	**.47**	**.32**	**.78**
Wiktionary	Les86	*.26*	*.21*	*.21*	*.17*	*.10*
Wiktionary-pseudo	Gur05	.50	.65	.35	.19	.24
Wikipedia	Les86	.38	*.24*	.26	*.09*	*.15*
Wikipedia-first	Les86	*.17*	*.17*	*.18*	*.12*	*.07*

(b) Results on German datasets.

Dataset		Gur–30	Gur–65	Gur–350
Word pairs used		22	39	115
		ρ	ρ	ρ
GermaNet-pseudo	Les86	.67	.68	.40
Wiktionary	Les86	*.02*	*.10*	*.01*
Wiktionary-pseudo	Gur05	**.75**	**.74**	**.45**
Wikipedia	Les86	*.04*	*.19*	.35
Wikipedia-first	Les86	*.02*	*.01*	.27

Table 6.2: Spearman correlation of gloss based measures with human judgments on English and German datasets. Best values for each dataset are in bold. Non-significant correlations are in italics (two tailed t-test, $\alpha = .05$).

6.2.2 Adapted Gloss Based Measures

Tables 6.2 (a) and (b) give an overview of the results of gloss based measures on the English and German datasets. On the English datasets, the gloss overlap measure based on WordNet pseudo glosses outperforms all other measures. On the German datasets, using the analogous GermaNet pseudo glosses also performs quite well but is slightly outperformed by using Wiktionary pseudo glosses. Wikipedia based glosses do not yield competitive results, because Wikipedia articles are less focused than WordNet or Wiktionary pseudo glosses which only contain the most important related concepts. Pseudo glosses outperform normal glosses by a wide margin for WordNet and Wiktionary, because normal glosses are usually quite short compared to the wealth of knowledge encoded in the relations used for constructing pseudo glosses.

6.2.3 Adapted Vector Based Measures

Tables 6.3 (a) and (b) display the results obtained for vector based measures. They generally yield relatively high values, but no resource clearly outperforms all others. On the German datasets, concept vectors based on Wiktionary entries yield the highest values, but on the Gur–350 dataset Wikipedia yields the same value and GermaNet is only slightly worse. WordNet and Wiktionary yield the best overall performance as they are also able to reliably estimate verb relatedness (YP–130 dataset). The WikipediaLink measure performs well on the Fin1–153 and Fin2-

(a) Results on English datasets.

	Dataset	MC–30	RG–65	Fin1–153	Fin2–200	YP–130
	Word pairs used	30	65	144	191	126
		ρ	ρ	ρ	ρ	ρ
WordNet	ZG07	.77	**.82**	.59	.48	**.73**
Wiktionary	ZG07	**.84**	.81	.67	**.54**	.63
Wikipedia	GM07	.72	.75	.67	.38	.29
Wikipedia-first	ZG07	.67	.73	**.68**	.51	.31
WikipediaLink	M07/NHN07	.45	.56	.60	.45	*.00*

(b) Results on German datasets.

	Dataset	Gur–30	Gur–65	Gur–350
	Word pairs used	24	50	203
		ρ	ρ	ρ
GermaNet	ZG07	.69	.70	.59
Wiktionary	ZG07	**.87**	**.86**	**.66**
Wikipedia	GM07	.80	.73	**.66**
Wikipedia-first	ZG07	.53	.57	.61
WikipediaLink	M07/NHN07	.58	.37	.36

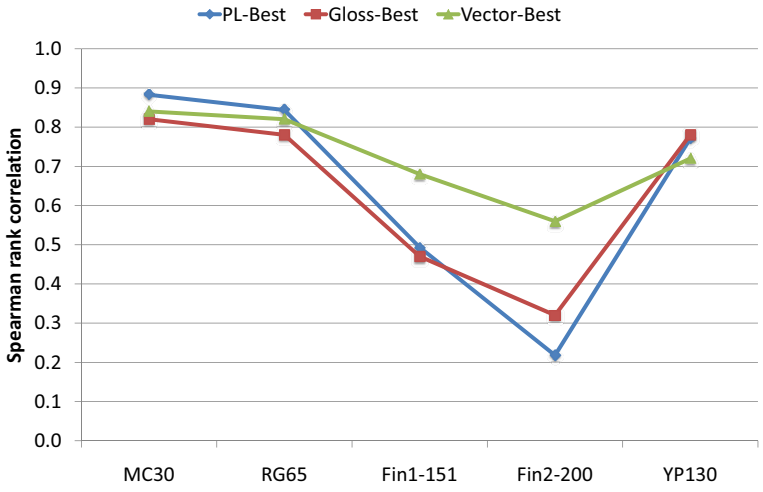
Table 6.3: Spearman correlation of vector based measures with human judgments on English and German datasets. Best values for each dataset are in bold. Non-significant correlations are in italics (two tailed t-test, $\alpha = .05$).

200 datasets. Our results are not directly comparable to the results in (Milne, 2007), as the measure was only evaluated on the full Fin–353 dataset and a different Wikipedia version was used for evaluation. However, Milne reports a Spearman correlation of .45 which is coarsely comparable to our results. The WikipediaLink measure is not able estimate classical relationships from the MC–30 and RG–65 dataset as well as the other vector based measures. It also completely fails to capture verb relatedness, which is to be expected as Wikipedia does not cover verbs well. Milne and Witten (2008a) report better performance using refined versions of the WikipediaLink measure. However, as they use a different Wikipedia version, results are not directly comparable.

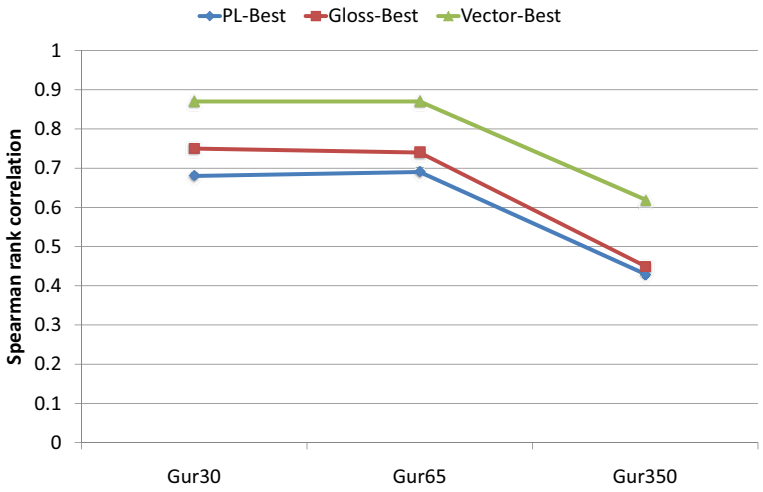
6.2.4 Comparison of Measure Types

So far, we have only compared semantic relatedness measures of a certain type using different semantic resources, but we have not compared measure types with each other. Figure 6.3 shows the maximum semantic relatedness value that is achieved by a particular measure type (path and IC based, gloss based, and vector based). For this analysis, we aggregated all measures of a certain type, and only show the best result for each measure type.

On the English datasets, all measure types perform comparably on the MC–30, RG–65, and YP–130 datasets. However, on the Fin1–153 and on the Fin2–200 datasets, the vector based measures outperform the other types by a wide margin. This is due to the fact, that the textual representation of a concept (e.g. a Wikipedia article text) contains a lot of additional information which is used by vector based



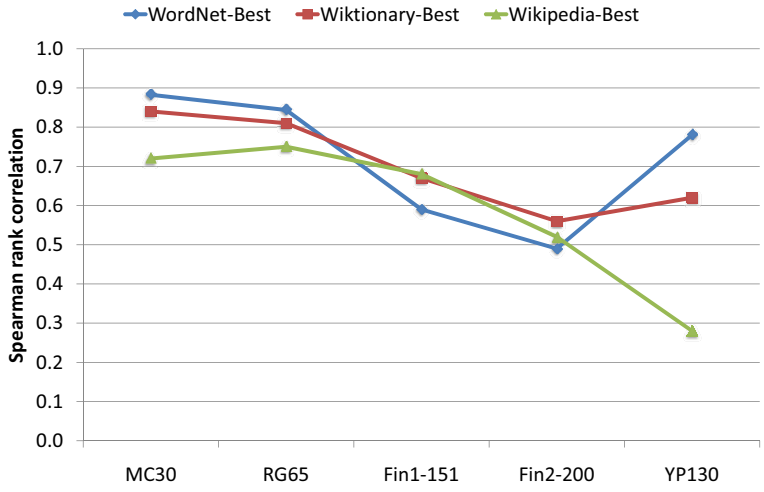
(a) English



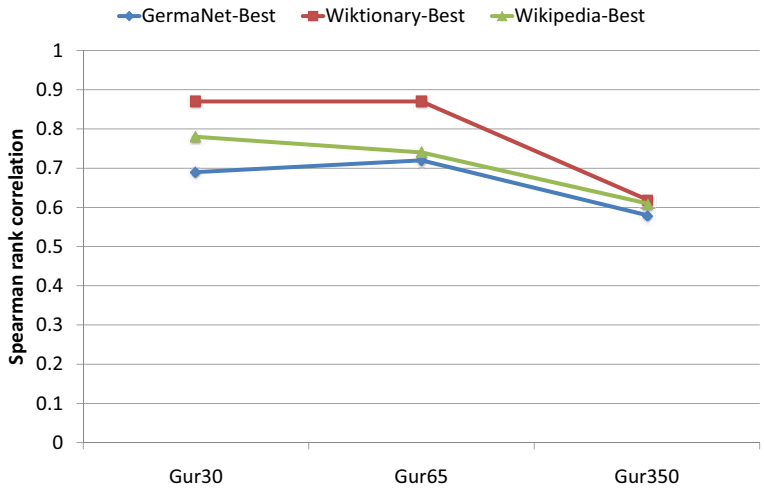
(b) German

Figure 6.3: Comparison of measure types.

measures to compute semantic relatedness. Gloss based measures also take the full article text into account, but only those of the two articles being compared, whereas vector based measures draw knowledge from *all* article texts in Wikipedia. For example, we analyzed the scores for the word pair (*Israel – Jerusalem*) which is highly related with a gold standard score of 8.46 on a 1–10 scale. The best path based measure did not consider the word pair as it is not covered by WordNet. The best gloss based measure yields a score of 0.0, i.e. there is no overlap between the glosses at all. The best vector based measure ranks the word pair very high (5th highest rank). Another example is the word pair (*abuse – drug*) with a medium to high gold standard score of 6.85. It gets very low scores from the path based and gloss based measures, while it is among the highest ranked word pairs according to the best vector based measure. Thus, we can conclude that concept vector based measures are better suited to estimate non-classical relationships as contained in the Fin1–153 and Fin2–200 datasets. This is verified on the German Gur–350 dataset that



(a) English



(b) German

Figure 6.4: Comparison of semantic resources.

also contains concepts connected via non-classical relationships. On that dataset, vector based measures also outperform the other measure types (see Figure 6.3 (b)). Interestingly, we find that vector based measures outperform the other measure types on all German dataset, which is in contrast to the findings on the English datasets. This is probably due to lower path and gloss based scores obtained using GermaNet as compared to using WordNet, because GermaNet is less developed than WordNet.

6.2.5 Comparison of Semantic Resources

Figures 6.4 (a) and (b) show the best obtained result for each semantic resource. Contrary to previous research (Strube and Ponzetto, 2006; Zesch et al., 2007b), we cannot draw the conclusion that CSRs like Wikipedia and Wiktionary are superior to LSRs like WordNet or GermaNet. On the two English datasets containing noun

Dataset		**Gur–65**	**Gur–65**	**Gur–350**
GermaNet	*PL/IC*	.97	.88	.36
	Gloss	.97	.88	.71
	Vector	.86	.77	.65
Wiktionary	*PL/IC*	-	-	-
	Gloss	.79	.71	.50
	Vector	.97	.94	.69
Wikipedia	*PL/IC*	.97	.94	.52
	Gloss	.97	.94	.52
	Vector	**1.00**	**1.00**	**.93**

Table 6.4: Coverage of semantic resources on German datasets. Best values for each dataset are in bold.

pairs connected by classical relations, WordNet performs best, closely followed by Wiktionary. On the verb similarity dataset (YP–130), WordNet performs much better than Wiktionary and Wikipedia, but Wiktionary at least yields mediocre results, while Wikipedia fails completely. The picture is different on the two datasets containing word pairs connected by non-classical relations (Fin1–153 and Fin2–200), where Wikipedia and Wiktionary slightly (but not statistically significant) outperform WordNet. For German, Wiktionary is the best resource, but on the Gur–350 dataset containing non-classically related and cross part-of-speech word pairs, all resources yield comparable results (see Figure 6.4 (b)). This means, that the performance on that dataset depends more on the measure type (see Figure 6.3 (b)) than on the semantic resource. Note that due to our experimental setup, we always evaluate on the subset of word pairs that is covered by all semantic resources used in certain experiment. Thus, coverage should also be considered when judging about the performance of a semantic resource and a certain semantic relatedness measure. We analyze coverage in the next section.

Comparison to Previous Results When comparing our best results with previously obtained values, we find that Patwardhan and Pedersen (2006) report slightly higher Spearman correlation values on the MC–30 and RG–65 datasets (.91 and .90), but the difference to our best results is not statistically significant. Gabrilovich and Markovitch (2007) report a Spearman correlation of $\rho = .75$ on a dataset consisting of Fin1–153 and Fin2–200. We were not able to reproduce this numbers. We obtained only $\rho = .67$ and $\rho = .38$ on the two subsets using a more recent Wikipedia version and a reimplementation of their method. Yang and Powers (2006) report a Pearson correlation coefficient of $r = .84$ for the YP–130 dataset using a path based measure specifically adapted to the verb taxonomy in WordNet. Their results cannot be directly compared to the Spearman rank correlation coefficient reported in this section. However, the full result tables in Appendix B show that we yield a comparable Pearson correlation of $r = .83$ using the Les86 gloss overlap measure with WordNet pseudo glosses.

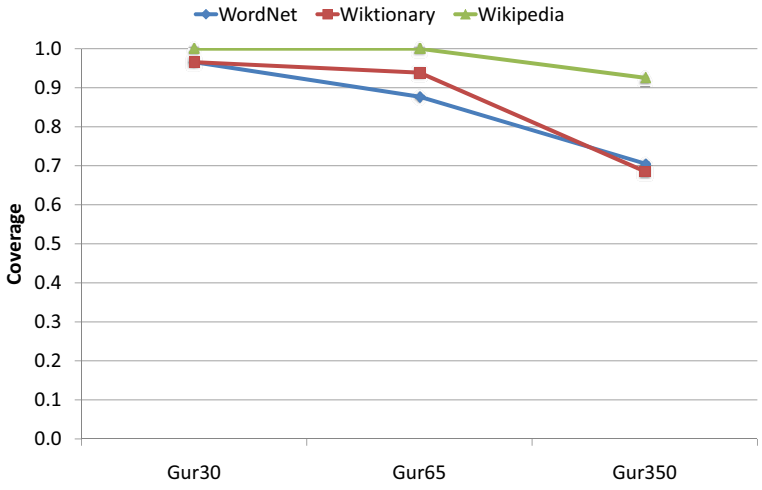


Figure 6.5: Coverage according to semantic resources on German datasets.

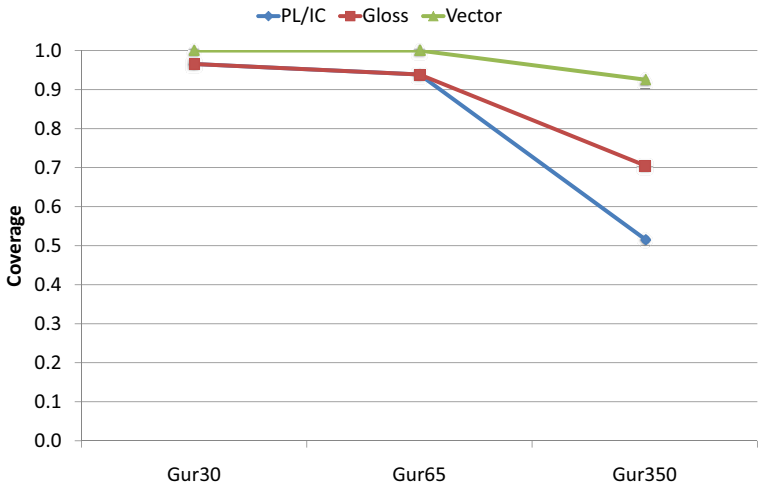


Figure 6.6: Coverage according to measure types on German datasets.

6.2.6 Coverage of Semantic Resources

For analyzing the coverage of semantic resources, we define a term to be *covered* by a semantic resource, if the term can be found in the resource. Hence, we define the coverage of a resource as the percentage of word pairs in a dataset where both terms are covered. Insufficient coverage of a resource is a major impediment for using semantic relatedness measures in large-scale natural language processing applications.

When analyzing coverage, we find that all *English* lexical-semantic resources including Wiktionary cover the datasets almost perfectly (coverage ranging from 97% to 100%). Hence, we only report detailed results on the German datasets in Table 6.4. The coverage of the German datasets is generally lower. This is due to the fact that German resources are not as well developed as the English resources. Another reason is that German datasets contain more domain specific word pairs which are not always covered by a general purpose knowledge source.

			Number of		
Date	**Name**	**Articles**	**Redirects**	**Categories**	**Disamb. pages**
01.12.2002	2002-2	8,596	658	0	0
01.06.2003	2003-1	19,236	2,574	0	0
30.11.2003	2003-2	37,999	9,397	0	0
30.05.2004	2004-1	93,930	24,379	0	0
28.11.2004	2004-2	173,837	51,765	4,180	17,682
29.05.2005	2005-1	246,113	81,198	11,176	23,571
27.11.2005	2005-2	338,887	126,050	19,114	30,157
28.05.2006	2006-1	434,211	177,413	24,591	39,019
26.11.2006	2006-2	537,868	240,271	31,936	50,113
27.05.2007	2007-1	641,178	333,657	39,158	64,898
25.11.2007	2007-2	727,186	404,431	45,889	71,522
25.05.2008	2008-1	815,609	477,790	52,385	78,051
23.11.2008	2008-2	895,136	547,244	59,453	83,798

Table 6.5: Growth of the German Wikipedia.

Figure 6.5 shows the values for each semantic resource aggregated over all measure types. We find that the coverage of German Wiktionary is comparable to the coverage of GermaNet even though Wiktionary has much less German word entries than GermaNet (see Section 2.3.2). On the Gur–350 dataset, Wikipedia's coverage outperforms the other resources by a wide margin. However, this is only because of the very high coverage of the vector based measures using Wikipedia that draw information from the full article texts. Thus, Figure 6.6 compares the coverage provided by the measure types aggregated over all semantic resources. While being comparable for all types on the Gur–30 and Gur–65 datasets, coverage of concept vector based measures outperforms gloss based and path length based measures by a wide margin on the domain-specific Gur–350 dataset.

6.2.7 Influence of Resource Growth

Conventional linguistic resources are rather static, as their structure is usually fixed and the content only changes from (infrequent) release to release. Collaboratively created resources are highly dynamic. For example in 2008, the English Wikipedia has grown by over 500,000 articles, i.e. about 1400 articles per day. Additionally, existing articles are augmented, new redirects are added, the category structure is changed, etc. Thus, in this section, we are going to investigate the influence of the growth of Wikipedia on its performance as a semantic resource for computing semantic relatedness.

As the Wikimedia Foundation also offers a Wikipedia dump that contains all revisions, we can reconstruct the state of the Wikipedia at any time since it was founded. For our experiments, we created a snapshot of the German Wikipedia every 183 days (6 months) starting December 1st, 2002. Table 6.5 shows the exact dates of the snapshots along with the number of articles, redirects, categories, and disambiguation pages in each snapshot. Figure 6.7 visualizes the growth of Wikipedia with respect to those snapshots. We evaluate the performance of semantic relatedness measures using each snapshot as a semantic resource. We limit the analysis to a careful selection of semantic relatedness measures from each measure

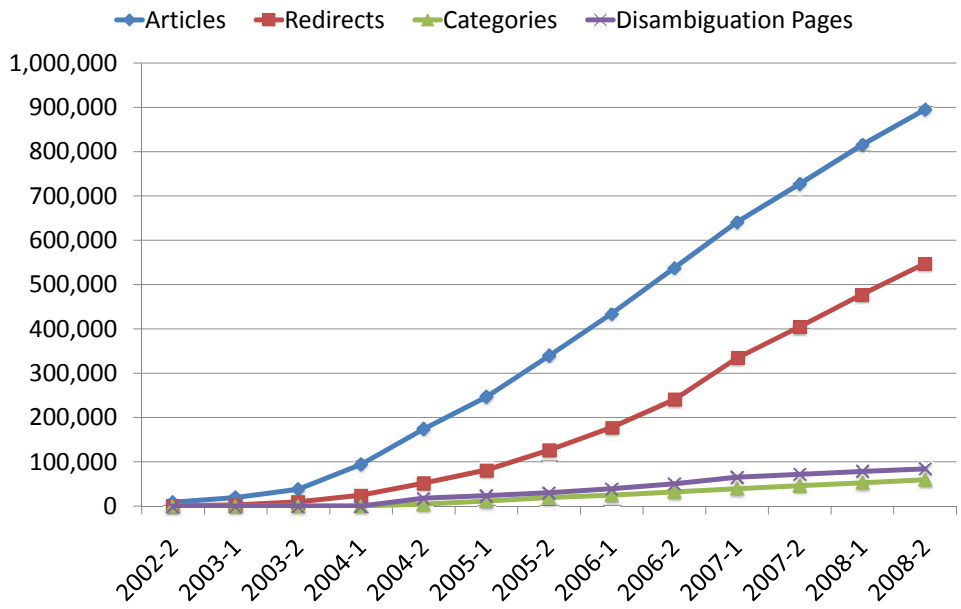


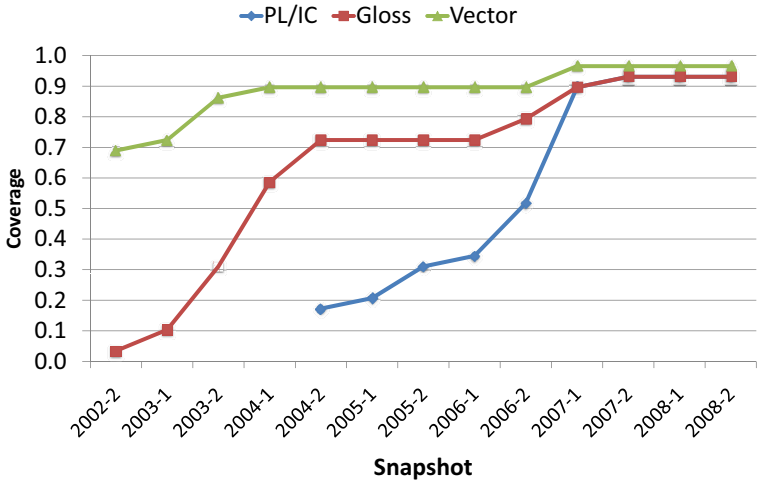
Figure 6.7: Growth of the German Wikipedia.

type. We select the Rad89 measure as the most versatile path based measure (called *PL/IC*), the gloss based Les86 measure, and the vector based measure GM07 using vectors built from the full Wikipedia articles.

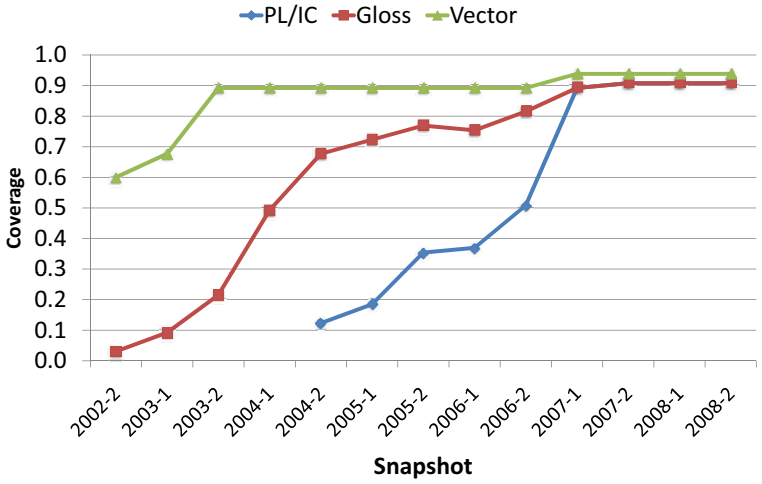
Figures 6.8 (a), (b), and (c) show the obtained coverage on the three German datasets. For the early snapshots, coverage rises steeply for all measure types, while for the recent snapshots coverage increases are small. We see that vector based measures generally cover more word pairs than the other measure types. Vector based measures also display high initial coverage even when using the quite small first snapshot from 2002. Path based measures do not cover any word pairs before the snapshot 2004-2, as they rely on the category system that was not added to Wikipedia until 2004. On the small Gur–30 and Gur–65 datasets, path and gloss based measures reach the same coverage as vector based measures using more recent snapshots. However, the results on the larger Gur–350 dataset in Figure 6.8 (c) show that vector based measures still have a much higher coverage than the other measure types.

Figures 6.9 (a), (b), and (c) show the obtained correlations on the three German datasets.[5] As correlation values based on a small number of word pairs are not reliable, we only present values where coverage reaches at least 20% of the full dataset and over 20 word pairs are covered. Thus, the lines in the chart corresponding to measure types with a low coverage do not cover all the snapshots. In general, the vector based measure shows the best performance. Only on two snapshots it is outperformed by the path based measure. However, the correlation values for these two snapshots are based on a small number of word pairs (as shown in Figure 6.8

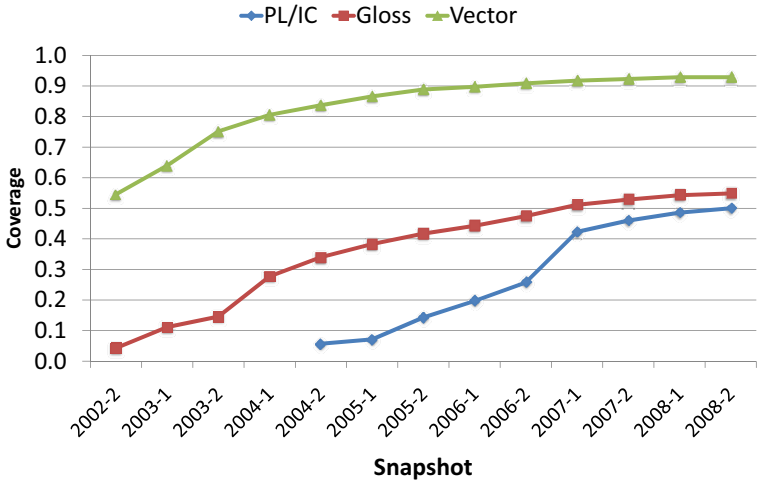
[5]The values in these charts are not directly comparable to the other results in this section. In other experiments, we limited the number of considered word pairs to those word pairs that are covered by all semantic relatedness measures being compared. In our growth analysis, we used all word pairs covered by a certain measure using a certain snapshot.



(a) Gur–30

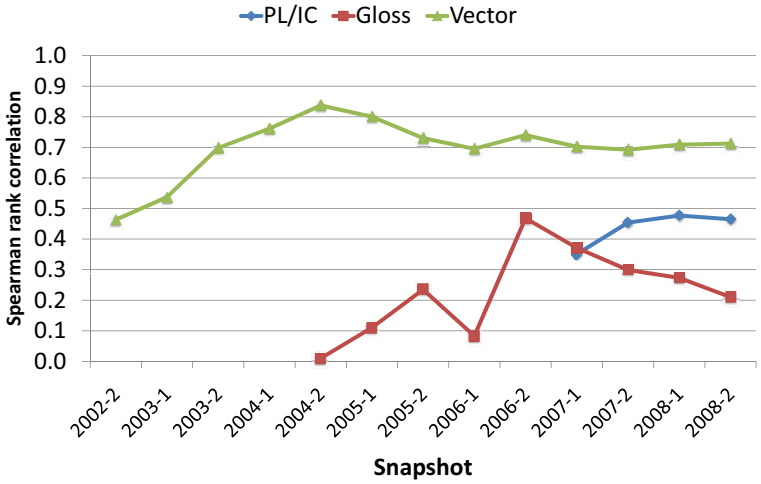


(b) Gur–65

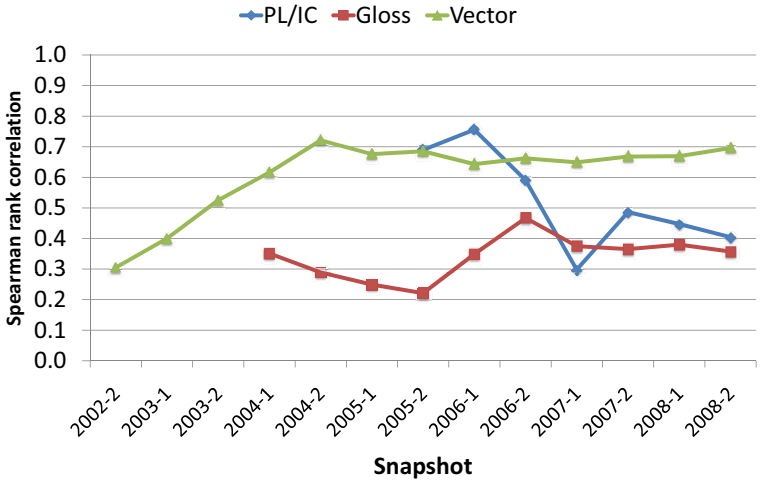


(c) Gur–350

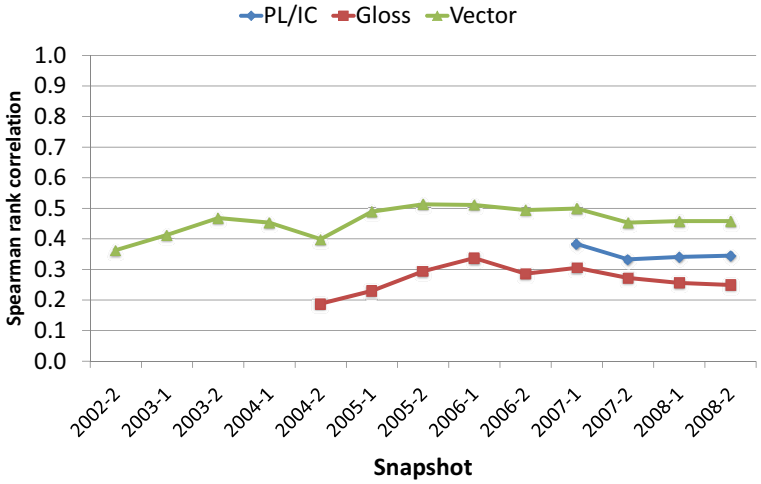
Figure 6.8: Influence of Wikipedia growth on the coverage of measures types.



(a) Gur–30



(b) Gur–65



(c) Gur–350

Figure 6.9: Influence of Wikipedia growth on the performance of measures types.

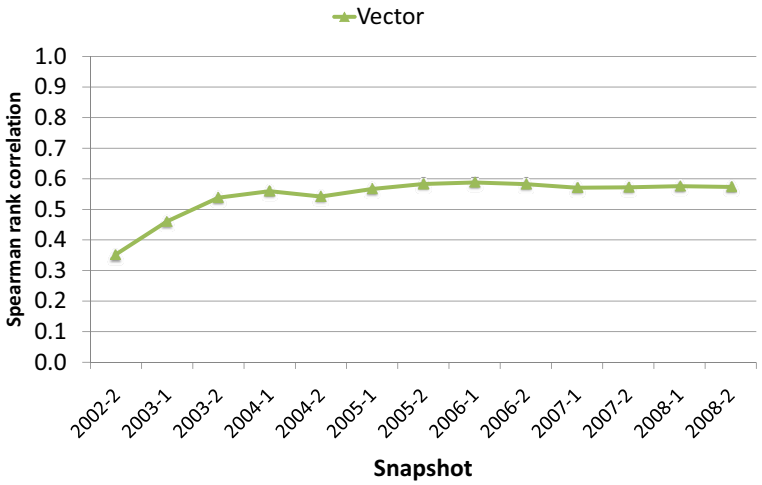


Figure 6.10: Influence of Wikipedia growth on the performance of vector based measures using a fixed set of word pairs.

(b)) and are quite unreliable. For the larger Gur–350 dataset in Figure 6.9 (c), we see rather stable distributions that do not show a clear rising or falling trend. This means that a growing Wikipedia leads to better coverage, but without the expected negative effects on the task performance.

However, in this analysis, the Spearman correlation scores are computed using as much word pairs as are covered by a certain snapshot. Thus, the analysis cannot tell us whether Wikipedia growth has an influence on the performance of semantic relatedness measures on the core set of word pairs covered by all snapshots. Thus, we perform an additional analysis where we only use a fixed number of word pairs covered by all snapshots. As we need a sufficient number of word pairs, we limit our analysis to the Gur–350 dataset and vector based measures. With this setting, even the initial snapshot from 2002 already covers more than 50% of all word pairs – cf. Figure 6.8 (c). Figure 6.10 visualizes the results. We see that, in the beginning, performance rises from snapshot to snapshot and then stays rather stable not showing a clear rising or falling trend. This means that even heavy changes like re-structuring, augmenting and adding articles, or addition of the category system do not decrease the performance on the initially covered word pairs. The vector based semantic relatedness measure seems to be remarkably stable in this respect.

Overall, we can conclude that – as expected – the growth of Wikipedia has a positive effect on coverage. Surprisingly, it has no or little negative effect on the suitability of Wikipedia for computing semantic relatedness. Especially for the vector based measure, correlation values and coverage are quite high even for smaller snapshots. Thus, even small language-specific versions of Wikipedia can be used for computing semantic relatedness in case there are no developed classical resources for a certain language. Another interesting conclusion is that if the initial coverage is already high enough for a certain task, smaller (and thus computationally less demanding) Wikipedia snapshots can be used without negative effects on the task performance.

(a) Results on English dataset.

	Measure	Type	Attempted	Score	# Ties	*Acc*	*Cov*	*H*
WordNet	Rad89	PL	199	127.8	24	**.64**	.69	**.66**
	WuP94	PL	196	123.3	17	.63	.68	.65
	LC98	PL	199	127.8	24	**.64**	.69	**.66**
	Res95	IC	196	120.5	48	.62	.68	.65
	JC97	IC	198	124.8	4	.63	.69	**.66**
	Lin98	IC	196	124.3	6	.63	.68	.65
Wikipedia	Rad89	PL	222	79.9	70	.36	**.77**	.49
	WuP94	PL	214	70.2	73	.33	.74	.46
	LC98	PL	222	79.9	70	.36	**.77**	.49
	Res95	IC	96	42.1	21	.44	.33	.38
	JC97	IC	222	54.5	21	.25	**.77**	.38
	Lin98	IC	222	52.5	201	.24	**.77**	.37

(b) Results on German dataset.

	Type	Measure	Attempted	Score	# Ties	*Acc*	*Cov*	*H*
GermaNet	PL	Rad89	294	186.8	22	.64	.30	.41
	PL	WuP94	238	158.7	16	.67	.24	.35
	PL	LC98	294	186.8	22	.64	.30	.41
	IC	Res95	137	127.0	0	**.93**	.14	.24
	IC	JC97	280	80.4	45	.29	.29	.29
	IC	Lin98	280	84.3	147	.30	.29	.29
Wikipedia	PL	Rad89	714	325.9	132	.46	**.73**	**.56**
	PL	WuP94	484	261.1	92	.54	.49	.51
	PL	LC98	714	325.9	132	.46	**.73**	**.56**
	IC	Res95	436	268.7	50	.62	.44	.51
	IC	JC97	713	323.3	45	.45	**.73**	**.56**
	IC	Lin98	436	263.6	35	.61	.44	.51

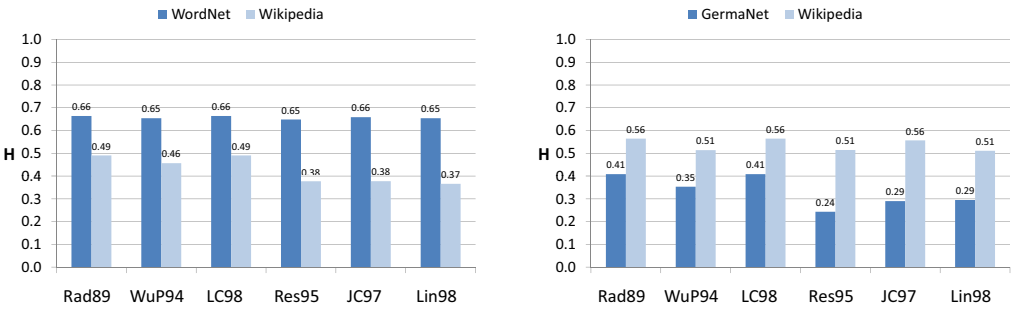
Table 6.6: Path based and IC based results on English and German word choice problems. Best values for each knowledge source are in bold.

6.3 Solving Word Choice Problems

In this section, we report the results obtained on the task of solving word choice problems. We first have a detailed look on each type of semantic relatedness measure, and then compare the performance yielded by individual measure types as well as semantic resources. We finally have a look at the coverage of semantic resources and at the influence of the growth of semantic resources on the results.

6.3.1 Adapted Path and IC Based Measures

Tables 6.6 (a) and (b) give an overview of the results of path and IC based measures on the English and the German dataset. We see that the measures tend to yield many ties. This is in general a problem for path-based measures, as the number of valid relatedness values is limited to discrete path length values. Figures 6.11 (a) and (b) visualize the performance of a certain measure on WordNet and GermaNet as compared with its adaptation to Wikipedia. Results differ between the English and German dataset. On the English dataset, path and IC based measures always



(a) English (b) German

Figure 6.11: Comparison of individual path based measures applied to WordNet, GermaNet, and Wikipedia.

(a) Results on English dataset.

Resource	Measure	Attempted	Score	# Ties	*Acc*	*Cov*	*H*
WordNet	Les86	280	137.5	73	.49	**.97**	.65
WordNet-pseudo	Gur05	182	148.8	10	.82	.63	**.71**
Wiktionary	Les86	273	86.2	64	.32	.95	.48
Wiktionary-pseudo	Gur05	31	29.5	1	**.95**	.11	.20
Wikipedia	Les86	223	63.0	6	.28	.77	.41
Wikipedia-first	Les86	223	59.6	64	.27	.77	.40

(b) Results on German dataset.

Resource	Measure	Attempted	Score	# Ties	*Acc*	*Cov*	*H*
GermaNet	Les86	197	169.3	3	**.86**	.20	.32
Wiktionary	Les86	201	77.9	21	.39	.20	.26
Wiktionary-pseudo	Gur05	101	87	0	**.86**	.10	.18
Wikipedia	Les86	714	266.3	16	.37	**.73**	.49
Wikipedia-first	Les86	694	279.0	49	.40	**.71**	**.51**

Table 6.7: Gloss based results on English and German word choice problems. Best values for each knowledge source are in bold.

work better when using WordNet than using Wikipedia. On the German dataset, it is the other way round. The reason for the different behavior seems to be that GermaNet is less developed than its English counterpart WordNet. However, we cannot compare the results directly as the English and the German dataset are of different difficulty for computational approaches introducing additional variability. The German dataset contains significantly more word choice problems using more complex domain-specific vocabulary that is more likely to be contained in Wikipedia than in GermaNet.

6.3.2 Adapted Gloss Based Measures

Tables 6.7 (a) and (b) display the results obtained for gloss based measures on the English and the German dataset. Wikipedia based glosses yield quite high coverage, but low accuracy on the English and the German dataset (Eng. $Acc = .28$,

(a) Results on English dataset.

Resource	Measure	Attempted	Score	# Ties	*Acc*	*Cov*	*H*
WordNet	ZG07	139	118.3	3	**.85**	.48	.61
Wiktionary	ZG07	156	128.3	3	.82	.54	.65
Wikipedia	GM07	280	144.0	2	.51	**.97**	**.67**
Wikipedia-first	ZG07	165	93.0	2	.56	.57	.56
WikipediaLink	M07/NHN07	141	69.8	3	.50	.49	.49

(b) Results on German dataset.

Resource	Measure	Attempted	Score	# Ties	*Acc*	*Cov*	*H*
GermaNet	ZG07	306	196.3	3	.64	.31	.42
Wiktionary	ZG07	310	276.8	2	**.89**	.32	.47
Wikipedia	GM07	814	584.5	4	.72	**.83**	**.77**
Wikipedia-first	ZG07	276	237.5	1	.86	.28	.42
WikipediaLink	M07/NHN07	383	245.5	4	.64	.39	.48

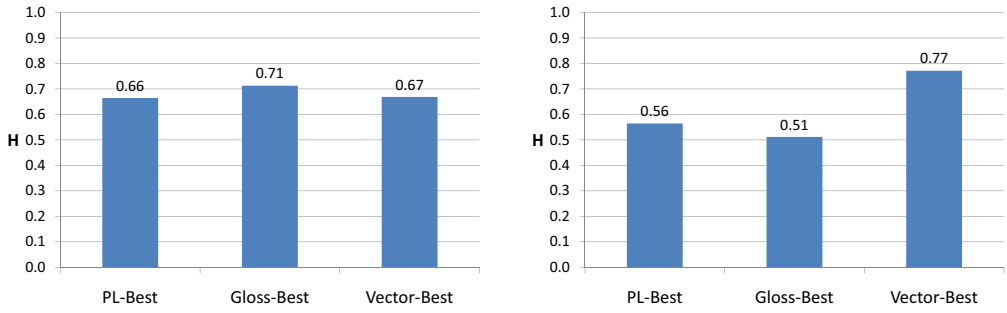
Table 6.8: Vector based results on English and German word choice problems. Best values for each knowledge source are in bold.

$Cov = .77$; Ger. $Acc = .37$, $Cov = .73$). The results for using the full Wikipedia article or just the first paragraph do not differ much, i.e. the first paragraph of a Wikipedia article is as informative for a gloss based measure as the whole article text. When comparing results for normal glosses with pseudo gloss results, we find that normal glosses give better coverage, while pseudo glosses give better accuracy. For example, using English Wiktionary glosses yields relatively low accuracy (.32) and almost perfect coverage of .95, while using English Wiktionary pseudo glosses yields almost perfect accuracy (.95) and very low coverage (.11).

6.3.3 Adapted Vector Based Measures

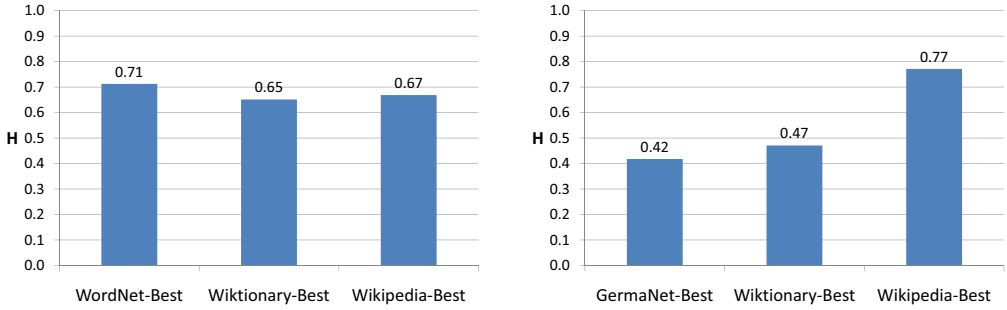
When looking at the overall performance values (harmonic mean H) in Tables 6.8 (a) and (b), we find that on the English dataset the vector based measures perform comparably using any of the resources ($H = .61 - .67$). However, using WordNet and Wiktionary yields very high accuracy, but only medium coverage, while it is the other way round when using Wikipedia ($Acc = .51$, $Cov = .97$). This is to be expected, as the full Wikipedia articles provide more textual information resulting in high coverage, but also contain non-relevant information which lowers accuracy. WordNet and Wiktionary contain more focused concept descriptions that provide high accuracy, but less coverage.

Important parameters of the concept vector based measure are the length and the quality of the textual representations used to create the vector space. Using the full Wikipedia article yields the best coverage (Eng. $Cov = .97$, Ger. $Cov = .83$) with reasonable accuracy (Eng. $Acc = .51$, Ger. $Acc = .72$). Using only the first paragraph yields higher accuracy (Eng. $Acc = .56$, Ger. $Acc = .86$), while the coverage is quite low in both cases (Eng. $Cov = .57$, Ger. $Cov = .28$). This is consistent with our previously described intuition, and allows us to configure the concept vector based measure according to whether high accuracy or high coverage



(a) English (b) German

Figure 6.12: Comparison of measure types.



(a) English (b) German

Figure 6.13: Comparison of semantic resources.

with reasonable precision is needed for an NLP application.

6.3.4 Comparison of Measure Types

So far, we have only compared semantic relatedness measures of a certain type using different semantic resources, but we have not compared measure types with each other. Figure 6.12 shows the maximum semantic relatedness value that is achieved by a particular measure type (path and IC based, gloss based, and concept vector based). For this analysis, we aggregated all measures of a certain type, and only show the best overall performance (H) for each measure type. We find that on the English dataset, all measure types perform comparably, while on the German dataset vector based measures clearly outperform the other measure types.

6.3.5 Comparison of Semantic Resources

Figures 6.13 (a) and (b) show the best obtained result for each semantic resource. We find that for the English word choice problems the best results are obtained using WordNet, but Wiktionary and Wikipedia perform comparably. In contrast to these findings, Wikipedia is much better suited to solve German word choice problems than GermaNet or Wiktionary. This is mainly due to the better coverage of Wikipedia in comparison to GermaNet or Wiktionary that we are going to analyze in the next section.

GermaNet	*PL/IC*	.30
	Gloss	.20
	Vector	.31
Wiktionary	*PL/IC*	-
	Gloss	.20
	Vector	.32
Wikipedia	*PL/IC*	.73
	Gloss	.73
	Vector	**.83**

Figure 6.14: Coverage of German semantic resources. Best values for each dataset are in bold.

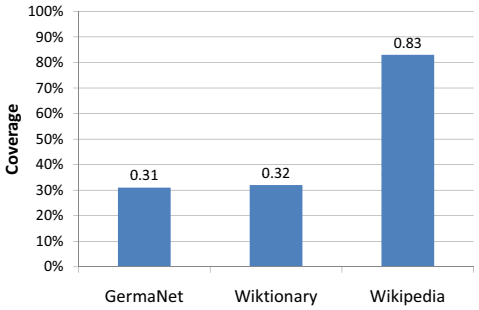


Figure 6.15: Coverage of semantic resources on German datasets.

To analyze the dependency between human and computational performance, we extracted the 50 easiest and the 50 most difficult word choice problems from the subset of the German dataset that was used to obtain human performance scores (see Section 5.2.1). Human performance on the easy problems is almost perfect ($Acc = .98$), while it drops to $Acc = .33$ on the difficult problems. The accuracy of computational methods is less affected as it drops only from .72 to .60 (obtained by the vector based measure using Wikipedia). Similarly, the coverage drops from .86 for the easy problems to .70 for the hard problems. This is due to the fact that hard problems contain more domain specific vocabulary that is not covered by the semantic resources.

We analyzed the results on the easy and hard word choice problems, and found that humans often fail if the candidate answers are similar with respect to spelling or pronunciation, or the candidate answers are strongly connotated with terms similar in spelling or pronunciation.[6] The performance of computational measures is not easily influenced by such distractors. Semantic relatedness measures are more likely to fail, if the candidate answers are all semantically related to the target. Humans also fail because they do not know the meaning of rare words used as candidate answers well enough, e.g. words from the medical ("Kinetose") or literature domain ("Distichon"). Computational measures are subject to a similar effect, as the evidence provided by a semantic resource may not be sufficient for rare words.

6.3.6 Coverage of Semantic Resources

Being able to solve word choice problems depends heavily on the coverage of a semantic resource. When analyzing coverage, we find that all *English* lexical-semantic resources cover the dataset almost perfectly (coverage ranging from 95% to 97%). Thus, we did not find much differences in the overall performance of English semantic resources in the previous section. Hence, we only report detailed results on the German dataset in Table 6.14. Figure 6.15 visualizes the coverage, but only shows the best value for each measure type. Wikipedia has a much higher coverage than GermaNet or Wiktionary. This also explains the large overall performance gains

[6] For example, one of the hardest problems was "antiquiert: a) ehrwürdig , b) alt, c) unbrauchbar, d) überholt". The correct answer is d). However, "antiquiert" sounds much like "Antiquariat" (antiquarian book-shop) misleading many humans to think that it means "alt" (old).

when using Wikipedia reported in the previous section for the German word choice problems.

6.3.7 Influence of Resource Growth

We are now going to investigate the influence of a growing Wikipedia on the performance on the task of solving word choice problems. We use the same setup as in Section 6.2.7: we use six monthly snapshots of the German Wikipedia from December 2002 to November 2008. Each snapshot is used as a semantic resource, and we compute the performance of semantic relatedness measures on this task. We limit the analysis to a careful selection of semantic relatedness measures from each measure type. We select the Rad89 measure as the most versatile path based measure, the gloss based Les86 measure, and the vector based measure GM07 using vectors built from the full Wikipedia articles.

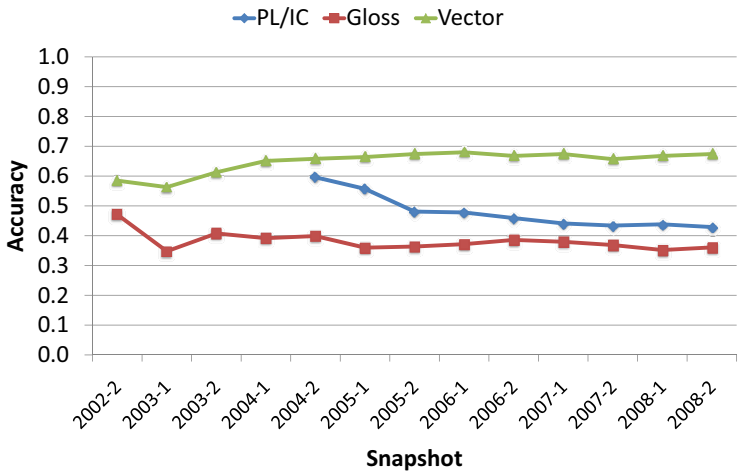
Figures 6.16 (a), (b), and (c) compare the three measure types according to accuracy, coverage, and harmonic mean of accuracy and coverage. We find that the accuracy values of vector and gloss based measures in Figure 6.16 (a) are almost stable for later snapshots, while the path based measure shows a falling trend. However, the higher values for the path based measure are unreliable, as they are obtained on snapshots with a very low coverage. Thus, we can conclude that the growth of Wikipedia has no negative effect on its suitability for solving word choice problems. However, it has a positive effect on coverage as shown in Figure 6.16 (b). Vector and gloss based measures display almost identical behaviour, but vector based measures have a higher coverage. Path based measures relying on the Wikipedia category graph only show comparable coverage for very recent snapshots. As accuracy is almost constant and coverage rises, the overall performance values (H) in Figure 6.16 (c) are bound to coverage.

6.4 Chapter Summary

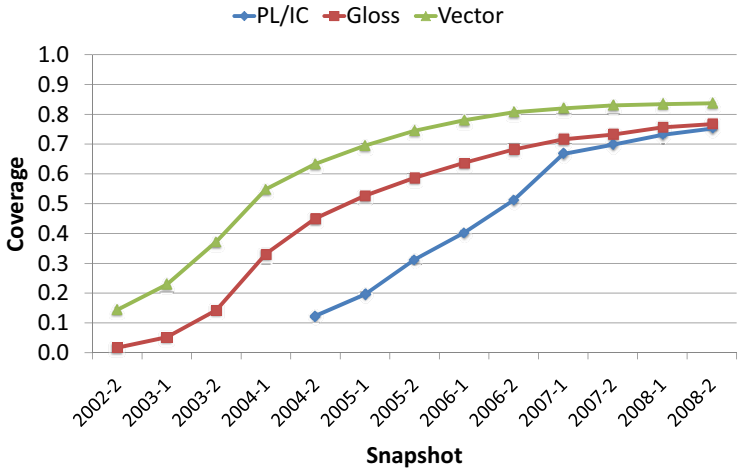
In this chapter, we presented a comprehensive evaluation of semantic relatedness measures. We analyzed the performance of these measures on two evaluation tasks *correlation with human judgments*, and *solving word choice problems* that we are going to summarize separately.

Correlation with human judgments

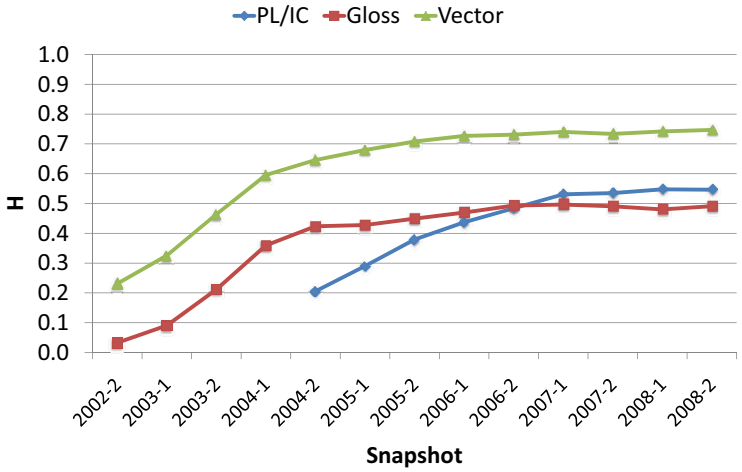
Measure adaptation We have shown that it is possible to adapt path based and information content based measures to Wikipedia, but the obtained results to not justify the efforts given the higher computational costs. We also have shown that gloss and vector based measures can be more easily adapted to all semantic resources which can be integrated into the interoperability framework. WordNet/GermaNet and Wiktionary yield the best performance for gloss based measures, while for vector based measures all considered semantic resources yield competitive results depending on the dataset.



(a) Accuracy



(b) Coverage



(c) Harmonic Mean – H

Figure 6.16: Influence of Wikipedia growth on solving word choice problems.

Measure types The vector based measure is the best measure type. On the German datasets, it always yields the best results. On the English datasets, it yields large performance increases on the two most complicated datasets containing non-classically related and cross part-of-speech pairs. On the other datasets, vector based measures perform comparably to the other measure types.

Semantic resources The performance reachable by a certain semantic resource depends on the evaluation dataset and thus on the kind of lexical-semantic phenomena that should be modelled. For English, WordNet yields the best performance on the verb relatedness datasets as well as on the two datasets containing noun pairs connected by classical lexical-semantic relations. Wikipedia and Wiktionary outperform WordNet on the two datasets containing also non-classical relations. For German, Wiktionary is the best resource, but differences are small when it comes to non-classically related and cross part-of-speech word pairs.

Coverage Coverage is almost perfect for the English datasets considered in this thesis. The coverage for German is generally lower, due to the resources being less developed. German Wikipedia displays a much higher coverage when compared with GermaNet and Wiktionary. GermaNet and Wiktionary show a comparable coverage which is remarkable given that the German Wiktionary is still quite small.

Resource growth Our analysis shows that the growth of Wikipedia has a positive effect on coverage. Surprisingly, it has no or little negative effect on the suitability of Wikipedia for computing semantic relatedness. Especially for vector based measures, correlation values and coverage are quite high even for smaller snapshots.

Solving word choice problems

Measure adaptation On the English dataset, path and IC based measures always perform better when using WordNet than using Wikipedia. On the German dataset, it is the other way round. The reason for the different behavior seems to be that GermaNet is less developed than its English counterpart WordNet.

Measure types We find that on the English dataset, all measure types perform comparably, while on the German dataset vector based measures clearly outperform the other measure types.

Semantic resources For the English word choice problems, the best results are obtained using WordNet, but Wiktionary and Wikipedia perform comparably. In contrast to these findings, Wikipedia is much better suited to solve German word choice problems than GermaNet or German Wiktionary. This is mainly due to the better coverage of the German Wikipedia.

Coverage We find that all English lexical-semantic resources cover the dataset almost perfectly (coverage ranging from 95% to 97%). The German Wikipedia has a much higher coverage than GermaNet or the German Wiktionary. This also explains

the large overall performance gains when using German Wikipedia as compared to GermaNet or German Wiktionary.

Resource growth We find that the growth of Wikipedia has a positive effect on its suitability for solving word choice problems. During growth, the accuracy is almost stable, while the coverage rises.

Conclusions

Overall, we found that collaboratively constructed semantic resources can fully substitute classical linguistically constructed semantic resources. When compared to large linguistically constructed semantic resources (e.g. WordNet for English), they yield comparable results, while for languages like German, with less developed classical resources, using collaboratively constructed semantic resources even leads to performance increases. Vector based semantic relatedness measures have shown to be the most versatile measure type showing good performance on a wide range of tasks and being easily applicable to all semantic resources.

When analyzing the influence of the growth of semantic resources (as examined on the example of the German Wikipedia), we find that it has no or little negative effect on the task performance, but – as expected – the coverage increases. Thus, collaboratively constructed semantic resources like Wikipedia can indeed be used as a proxy for linguistically created semantic resources that might not exist for minor languages.

In this chapter, we focused on an intrinsic evaluation to directly examine the properties of semantic relatedness measures while minimizing other influences. Finally, a semantic relatedness measure should be tested inside the scope of an application to prove that the improved performance has a positive impact on the application. In the next chapter, we focus on keyphrase extraction as a possible application of semantic relatedness measures.

Chapter 7

Using Semantic Relatedness to Enhance NLP

In the previous chapter, we presented the results of intrinsically evaluating semantic relatedness measures. We found that the performance of measures differs with respect to measure type and semantic resource. However, we still do not know if these differences have an impact on real-life applications. Thus, in this chapter, we analyze the performance of semantic relatedness measures in a real-life application scenario. We first introduce in Section 7.1 the task of keyphrase extraction, and show how semantic relatedness measures can be applied to this task. In Section 7.2, we then briefly describe other applications in which semantic relatedness measures have been applied: semantic information retrieval and context-aware user interfaces. We conclude with a chapter summary in Section 7.3.

7.1 Keyphrase Extraction

Keyphrases are small sets of expressions representing the content of a document. **Keyphrase extraction** is the task of automatically extracting such keyphrases from a document. The extracted phrases have to be present in the document itself, in contrast to keyphrase assignment (a multi-class text classification problem) where a fixed set of keyphrases is used that are not necessarily contained in the document. Keyphrase extraction has important applications in NLP including summarization (D'Avanzo and Magnini, 2005; Litvak and Last, 2008), clustering (Hammouda et al., 2005), highlighting (Turney, 2000), searching (Bracewell et al., 2005), or indexing and browsing (Gutwin et al., 1999).

In Section 7.1.1, we give an overview of state-of-the-art approaches to keyphrase extraction. In Section 7.1.2, we propose a new approach to keyphrase extraction based on computing the semantic relatedness between terms in a document. In Section 7.1.3, we introduce a generalized keyphrase extraction framework, and in Section 7.1.4, we describe our evaluation setup. We analyze the experimental results in Section 7.1.5 and conclude with a summary in Section 7.1.6.

7.1.1 State-of-the-Art

State-of-the-art methods for keyphrase extraction can be categorized into supervised and unsupervised approaches. Supervised approaches require a manually annotated corpus for each target domain. Unsupervised approaches do not require any training data, but their performance is usually lower.[1] Closely related to the field of keyphrase extraction are glossary extraction (Park et al., 2002) and back-of-the-book indexing (Csomai and Mihalcea, 2007).

Unsupervised Approaches

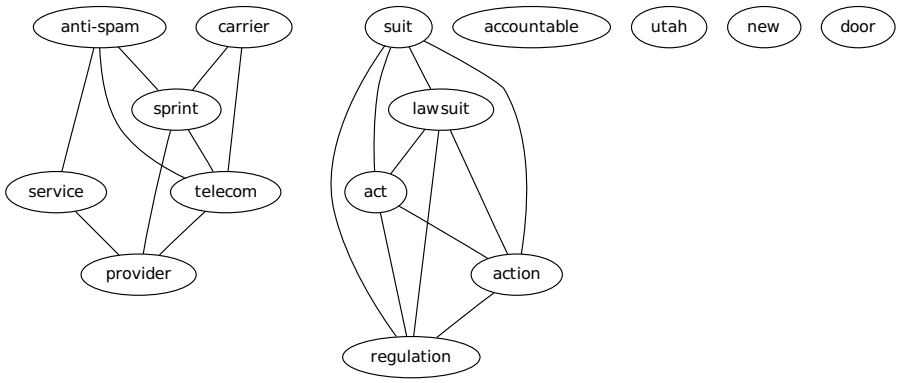
Unsupervised approaches usually select quite general sets of candidates (like all noun phrases or all tokens in a document), and use a subsequent ranking step to limit the selection to the most important candidates. For example, Barker and Cornacchia (2000) restrict candidates to noun phrases, and rank them using heuristics based on length, term frequency, and head noun frequency. Bracewell et al. (2005) also restrict candidates to noun phrases, and cluster them if they share a term. The clusters are ranked according to the noun phrase and token frequencies in the document. Finally, the centroids of the top-n ranked clusters are selected as keyphrases. Mihalcea and Tarau (2004) propose a graph-based approach called *TextRank*, where the graph nodes are tokens and the edges reflect co-occurrence relations between tokens in the document. The nodes are ranked using PageRank (Page et al., 1999), and longer keyphrases can be reconstructed in a post-processing step merging adjacent keywords. The method was found to yield competitive results with state-of-the-art supervised systems. Wan and Xiao (2008) expand *TextRank* by augmenting the graph with highly similar documents, which improves results compared with standard *TextRank* and a tf.idf baseline.

Another branch of unsupervised approaches is based on statistical analysis. Tomokiyo and Hurst (2003) use pointwise KL-divergence between language models derived from the documents and a reference corpus. Paukkeri et al. (2008) use a similar method based on likelihood ratios. Matsuo and Ishizuka (2004) present a statistical keyphrase extraction approach that does not make use of a reference corpus, but is based on co-occurrences of terms in a single document.

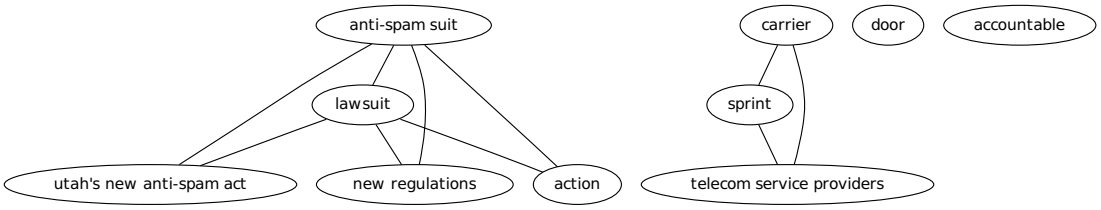
Supervised Approaches

Supervised approaches use a corpus of training data to learn a keyphrase extraction model that is able to classify candidates as keyphrases. A well known supervised system is *Kea* (Frank et al., 1999) that uses all n-grams of a certain length as candidates, and ranks them using the probability of being a keyphrase. *Kea* is based on a Naïve Bayes classifier using *tf.idf* and *position* as its main features. *Extractor* (Turney, 2000) is another supervised system that uses stems and stemmed n-grams as candidates. Its features are tuned using a genetic algorithm. *Kea* and *Extractor* are known to achieve roughly the same level of performance (Turney,

[1] Note that unsupervised approaches might use tools like noun phrase chunkers relying on supervised approaches which require training data. However, as such tools are usually already available for most languages, we consider an approach to be unsupervised if it does not make use of any training data with annotated *keyphrases*.



(a) Lexical-semantic graph based on lemmas.



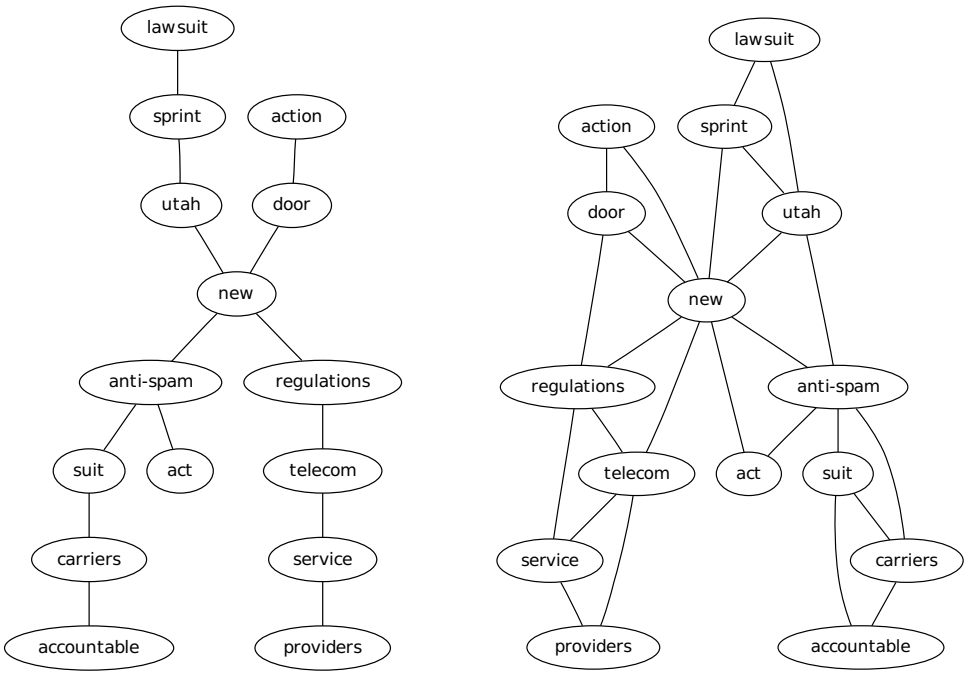
(b) Lexical-semantic graph based on noun phrases.

Figure 7.1: Example document represented as a lexical-semantic graph. We only show edges representing strong relationships.

2003). Hulth (2003) uses a combination of lexical and syntactic features adding more linguistic knowledge which outperforms *Kea*. Medelyan and Witten (2006) present the improved *Kea++* that selects candidates with reference to a controlled vocabulary from a thesaurus or Wikipedia (Medelyan et al., 2008). Turney (2003) augments *Kea* with a feature set based on statistical word association to ensure that the returned keyphrase set is coherent. However, this assumption might not hold if a document covers different topics. Nguyen and Kan (2007) augment *Kea* with features tailored towards scientific publications such as section information and certain morphological phenomena often found in scientific papers.

7.1.2 Lexical-Semantic Graphs

We propose a new approach to keyphrase extraction that is based on measuring the semantic relatedness between terms in a document. Most other approaches use a term's frequency as an important clue to decide whether the term is a keyphrase or not, but due to reading and writing economy, terms might not be repeated in a document (Barker and Cornacchia, 2000). However, even a term that occurs only once in a document might be of high importance if it is semantically related to many other terms in the same document. To find such important concepts, we construct a *lexical-semantic graph* (LSG). An LSG is a fully connected undirected graph $G = (V, E)$ where a node v_i represents a term in the document, and the weight w of an edge $\{v_i, v_j\}$ represents the strength of the semantic relatedness between the terms. The nodes are ranked using PageRank (Page et al., 1999), i.e. the most



(a) Context window of size 1. (b) Context window of size 2.

Figure 7.2: Example document represented as a token based co-occurrence graph.

central nodes in the graph will be selected as keyphrases. The approach is similar to the co-occurrence graph based extraction approach (Mihalcea and Tarau, 2004), but in an LSG, edges represent the strength of semantic relatedness between two words, while in a co-occurrence graph an edge represents the co-occurrence of two words in the document. To visualize the difference, we take an example document from the Inspec dataset:

> Anti-spam suit attempts to hold carriers accountable.
> A lawsuit alleges that Sprint has violated Utah's new anti-spam act. The action could open the door to new regulations on telecommunication service providers.

and represent it as a lexical-semantic graph (see Figure 7.1) and as a co-occurrence graph (see Figure 7.2). As the lexical-semantic graph is a fully connected graph that cannot be easily visualized, we only show edges representing strong relationships. We also assume for the sake of the example that the LSG was created using a semantic relatedness measure which perfectly determines the relationships between the words in the document. The resulting LSG would then contain two clusters corresponding to *lawsuit* related words and *telecommunication* related words as in Figure 7.1. The keyphrases for that document will be selected with high probability from both clusters thus covering both topics. In contrast to the LSG, the co-occurrence graph (see Figure 7.2) is less structured, as there is not much repetition of words in the document. It centers around the word "new" that co-occurs with many other words, but is rather unimportant for the document.

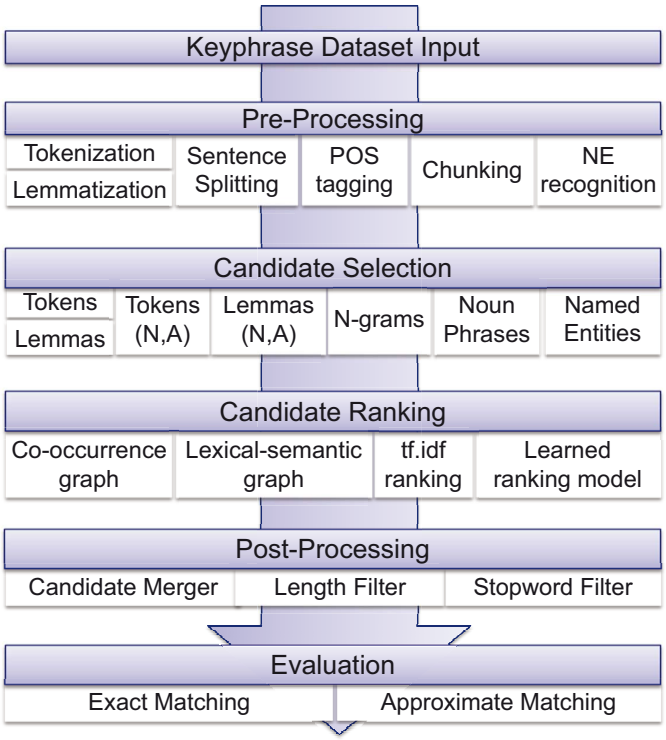


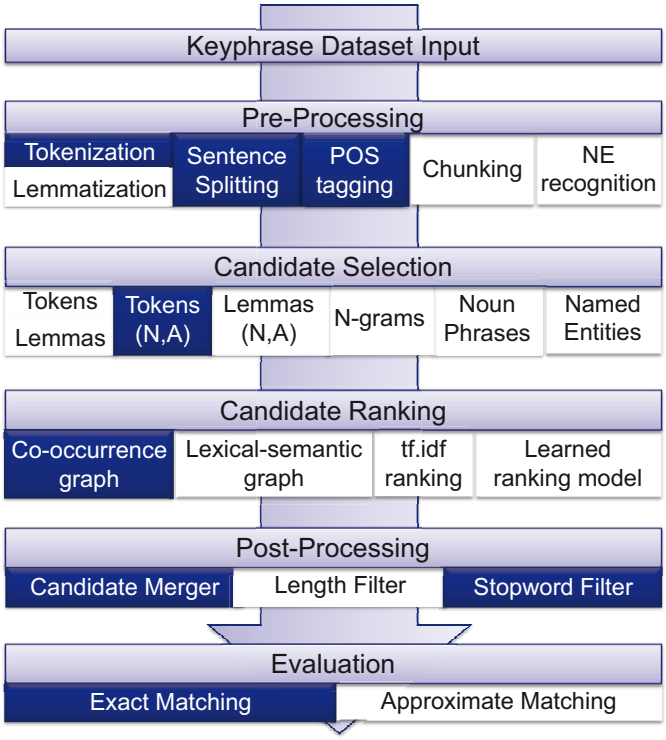
Figure 7.3: Overview of the keyphrase extraction framework.

7.1.3 Keyphrase Extraction Framework

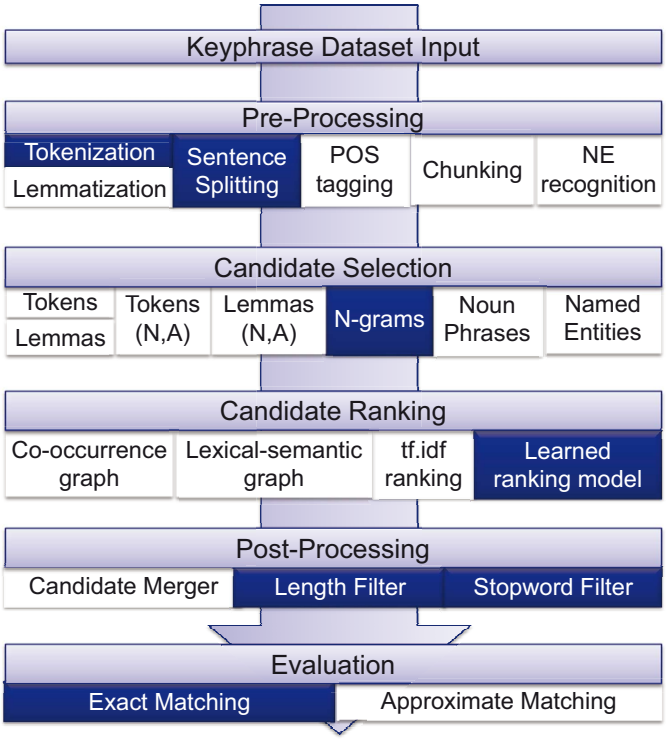
Most automatic keyphrase extraction methods have two stages: first they select a list of keyphrase candidates that is then ranked according to some measure of keyphrase importance. To allow for a fair comparison of different keyphrase extraction approaches, the same pre- and postprocessing should be applied, as well as exactly the same evaluation strategy. We propose a generalized framework for the comprehensive analysis of keyphrase extraction as shown in Figure 7.3. It was designed to be as language-independent as possible, with components either using no language dependent information at all, or components that are already available for most languages (like tokenizers or chunkers). Figures 7.4 (a) and (b) show how the state-of-the-art keyphrase extraction approaches TextRank (Mihalcea and Tarau, 2004) and Kea (Frank et al., 1999) are modelled in the framework.

Preprocessing and Candidate Selection

For preprocessing, we tokenize the documents, and split them into sentences. We integrated the TreeTagger for lemmatization, part-of-speech tagging, and noun phrase chunking (Schmid, 1995), as well as the Stanford NER (Finkel et al., 2005) for named entity recognition. From this pool of preprocessed data, we select as candidates **Tokens**, **Lemmas**, **N-grams**, **Noun Phrases**, and **Named Entities**. As the *TextRank* keyphrase extraction system (Mihalcea and Tarau, 2004) restricts candidates to nouns and adjectives, we additionally use the restricted set of tokens **Tokens (N,A)** and lemmas **Lemmas (N,A)**.



(a) TextRank (Mihalcea and Tarau, 2004)



(b) Kea (Frank et al., 1999)

Figure 7.4: State-of-the-art keyphrase extraction systems represented in the proposed framework.

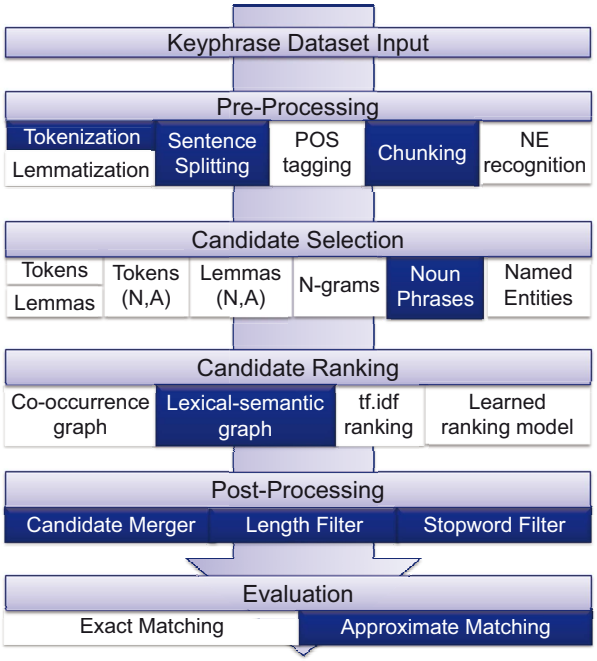


Figure 7.5: Lexical-semantic graph keyphrase extraction approach represented in the framework.

Candidate Ranking

The co-occurrence graph-based *TextRank* method builds a co-occurrence graph using the keyphrase candidates. The final candidate ranking is determined by computing the centrality scores of the graph nodes using PageRank. For tf.idf ranking, the tf.idf scores are computed using token frequencies. If candidates contain more than one token, we set the overall tf.idf score to the maximum tf.idf score among all the contained tokens. The supervised keyphrase extraction systems use the extraction model obtained from the training data to classify the candidates into keyphrases and rank them according to their importance in the document.

For our LSG approach, we build a fully connected graph from the candidates, and set the weights of the edges according to the semantic relatedness between the nodes. The semantic relatedness measure is selected from those presented in Chapter 3. The final candidate ranking is determined by computing the centrality scores of the graph nodes using PageRank.

Postprocessing and Evaluation

We merge candidates that are adjacent in the source document, as some keyphrase extraction systems (e.g. *TextRank*) use single term candidates like *Tokens*, and rely on a subsequent merging step to reconstruct longer keyphrases. However, to ensure a fair comparison we apply merging to all keyphrase extraction systems, because also approaches with higher quality candidates like noun phrases can benefit from merging. For example, the two noun phrases “improved scheduling” and “algorithm” could be merged to “improved scheduling algorithm” if they are adjacent in the document.

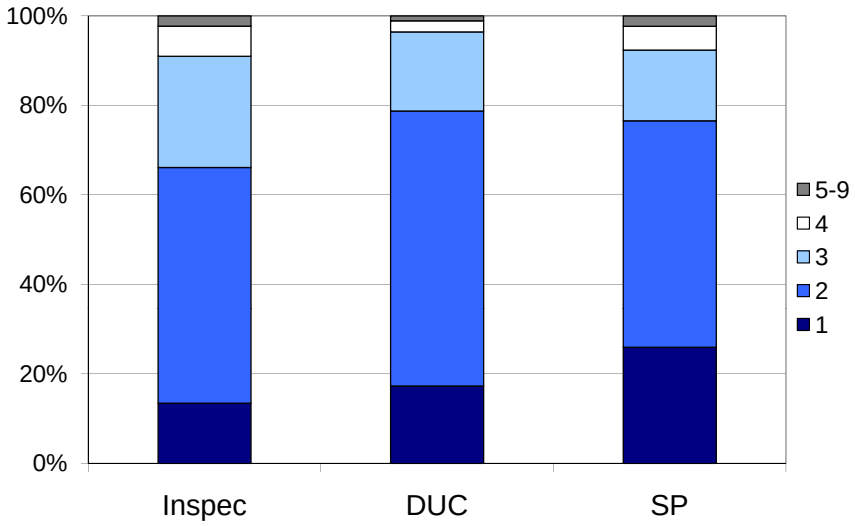


Figure 7.6: Number of tokens per keyphrase.

We use an additional post-filtering step to remove candidates or keyphrases that do not conform to length restrictions. When analyzing the length of the gold standard keyphrases in the training set, we found that - depending on the dataset - 97,7% to 99,2% of all keyphrases in the training data contain 1 to 4 tokens (see Figure 7.6). For that reason, we limited the length of returned keyphrases to 1 to 4 tokens.

We remove trailing stopwords from candidates, but keep stopwords that appear inside a candidate. For example, we keep "United States of America" as the stopword appears inside a candidate, while "the weak economy" is pruned to "weak economy" as the stopword occurs at the boundary of the candidate. We also remove candidates from the candidate list if they exactly match a stopword.

Finally, the post-processed list of ranked keyphrases candidates is used to evaluate the keyphrase extraction systems. We describe the evaluation setup in the next section.

7.1.4 Evaluating Keyphrase Extraction

The prevalent approaches for evaluating keyphrase extraction algorithms are: (i) *manual evaluation based on human judges* (Barker and Cornacchia, 2000; Turney, 2000; Matsuo and Ishizuka, 2004), (ii) *application-based evaluation* (Bracewell et al., 2005; Litvak and Last, 2008), and (iii) *automated evaluation against human assigned keyphrases* (Frank et al., 1999; Turney, 2003; Hulth, 2003; Mihalcea and Tarau, 2004; Nguyen and Kan, 2007).

In **manual evaluation**, human judges decide whether the returned keyphrases are good representatives of a document's content or not. Thus, this evaluation approach is not restricted to exact matches between gold standard keyphrases and keyphrases returned by a method. However, manual evaluation of extracted keyphrases is very costly and time-consuming. In particular, it is not suited for any kind of parameter tuning, as the output of each new system configuration involves manual re-evaluation.

An **application-based evaluation** utilizes keyphrases as part of a usually complex application, and the performance is measured in terms of the overall performance of the application. However, this entails influence of parameters besides the keyphrase extraction algorithm to be tested. For example, Bracewell et al. (2005) use the information retrieval task of keyword search to determine the effectiveness of keywords at uniquely describing the document from which they were extracted. However, this method might extract keyphrase sets that are good indicators for relevant documents, but that are not acceptable when presented to humans. Litvak and Last (2008) use a summary-based evaluation, where a term is used as a gold standard keyphrase if it appears in the document and in the summary.

Automated evaluation against human assigned keyphrases relies on automated matching of human annotated gold standard keyphrases with the keyphrases extracted by a certain approach. The human assigned keyphrases are either derived from keyphrases assigned by authors (Frank et al., 1999; Turney, 2000), or are annotated by indexers (Hulth, 2004; Nguyen and Kan, 2007; Wan and Xiao, 2008). As this approach avoids the problems of manual evaluation (costly, time-consuming, difficult algorithm tuning), and of application-based evaluation (influence of complex applications, keyphrases unacceptable to humans), we are going to use it for our evaluation.

Despite the importance of the task, the automated evaluation of keyphrase extraction has not received much research attention in the past. We address three core problems with the automated evaluation of keyphrase extraction: (i) the evaluation datasets, (ii) the evaluation metric, and (iii) the evaluation framework.

Evaluation datasets Comparing results from different papers is difficult as no standard datasets are used and very few papers have compared their results on more than one dataset with different competing systems. Thus it cannot be judged conclusively which approaches improve results on which kind of dataset. We collected three publicly available datasets with different properties, which allows to compare the applicability of keyphrase extraction algorithms to those datasets.

Evaluation metric The performance of most keyphrase extraction algorithms is evaluated by comparing whether the extracted keyphrases exactly match the human assigned gold standard keyphrases. However, this is known to underestimate performance (Turney, 2000). Allowing only exact matching cannot account for variations in the extracted keyphrases that might be perfectly acceptable when presented to humans. For example, longer noun phrases like "congress party spokesman" are usually more specific and thus more informative to the reader than shorter noun phrases like "congress party". However, due to reading and writing economy, specific words are usually not often repeated in a document (Barker and Cornacchia, 2000). Thus, longer noun phrases are unlikely to be marked by human annotators which prevents exact matching. To compensate for these shortcomings, we propose a new approximate matching strategy that also accounts for non-exact matches, and is able to give a better picture of the actual quality of a keyphrase extraction algorithm.

Dataset	Domain	# Indexers	# Docs	∅ # Tokens	∅ # Keyphrases	r
Inspec	Scientific	Single	2000	138.6	9.64	0.56
DUC	News	Multiple	301	902.8	8.08	0.18
SP	Scientific	Multiple	134	8491.6	8.31	0.08

Table 7.1: Keyphrase evaluation datasets. r is the Pearson correlation between the document length and the number of assigned keyphrases.

Evaluation framework Some datasets contain annotated keyphrases that actually cannot be found in the document. This has serious implications on the comparability of results, as including them in the evaluation might significantly lower the reachable performance on the dataset. A way to solve this problem is to use a unified framework for the evaluation of keyphrase extraction. This also prevents influence from varying pre- and postprocessing. Thus, to ensure fair testing conditions, we use the generalized keyphrase extraction framework as described in Section 7.1.3.

Evaluation Datasets

We now describe three publicly available datasets with manually annotated gold standard keyphrases. They differ in length and domain (see Table 7.1), and can thus be used to assess different properties of keyphrase extraction algorithms.

The **Inspec dataset** (Hulth, 2004) contains 2000 abstracts of journals in the Inspec database from the years 1998 to 2002.[2] There are two sets of keyphrases assigned by professional indexers: controlled terms (restricted to the Inspec index terms, and useful for keyphrase assignment) and uncontrolled terms. Some uncontrolled terms (23.8%) are not directly found in the documents and therefore ignored in our evaluation. However, this dataset has the highest number of human assigned keyphrases per document, while the documents are rather short with an average length of ≈ 140 tokens. The correlation between the length of the document and the number of human assigned keyphrases is quite high (Pearson correlation $r = 0.56$), indicating that indexers often exhaustively annotated keyphrases in the documents. Thus, it should be relatively easy to extract keyphrases from the documents, and we expect the performance on this dataset to be higher than on the other datasets.

The **DUC dataset** (Wan and Xiao, 2008) consists of 308 documents from DUC2001 that were manually annotated with at most 10 keyphrases per document by two indexers. Annotation conflicts between the indexers were solved by discussion. Two documents in the DUC2001 data obtained from NIST[3] were empty, and 5 documents had no annotated keyphrases. Thus, the final dataset used in this thesis contains 301 documents.

The **SP dataset** (Nguyen and Kan, 2007) originally contains 211 scientific publications downloaded from the internet that were automatically converted to plain text. Keyphrases were manually annotated by multiple indexers, but conflicts were not resolved. We removed documents for which no keyphrase annotation was available, and those with multiple conflicting annotations. The final dataset contains

[2] `http://www.theiet.org/publishing/inspec/`

[3] `http://duc.nist.gov/`

134 documents.

As manually creating such datasets is still costly, Paukkeri et al. (2008) propose to use Wikipedia, which is available for a wide range of languages. They use links in Wikipedia articles as a substitute for human keyphrase annotations, similar to the task of reproducing links in Wikipedia (Csomai and Mihalcea, 2007; Milne and Witten, 2008b). The assumption is that links in Wikipedia reflect keyphrases that would have been selected for that document. However, links in Wikipedia fulfill a wide range of functions including navigation and reference. Additionally, linking might be restricted due to a maximum number of allowed links per sentence, non-existing articles, or important concepts just being highlighted instead of linked. Thus, we do not consider this evaluation approach in our study.

Evaluation Metric

Automated evaluation of keyphrase extraction relies on matching a set of human annotated gold standard keyphrases K_{gold} with a ranked list of keyphrases K_{ext} extracted by a certain approach. We define a matching m between a gold standard keyphrase $k_{gold} \in K_{gold}$ and an extracted keyphrase $k_{ext} \in K_{ext}$ to be a tuple $m = (k_{gold}, k_{ext})$. The matching can either be true or false, depending on whether k_{gold} and k_{ext} are equivalent according to the matching strategy. Previous works used exact matching (EXACT) that requires k_{gold} and k_{ext} to have exactly the same string representation, i.e. $\text{EXACT}(k_{gold}, k_{ext}) = \text{true} \Leftrightarrow k_{gold} = k_{ext}$.

To evaluate the overall performance of a keyphrase extraction system, we do not need to look at single matchings m, but at the full list of matchings M. Previous studies used Precision (P), Recall (R), and F-measure (F_1) at a certain fixed cutoff value, e.g. after the first 10 retrieved keyphrase matchings. However, if documents have varying numbers of keyphrases assigned (which is the case for all datasets presented in Section 7.1.4), a cutoff might distort results for some documents. For example, if we always extract 10 keyphrases, but a document only has 8 gold keyphrases assigned, then 2 extracted keyphrases will always be wrong. Thus, we propose to use the **R-precision (R-p)** measure from information retrieval to evaluate keyphrase extraction systems. In information retrieval, R-p is the precision when as many documents have been retrieved as relevant documents are in the document collection. Hence, for keyphrase extraction R-p is defined as the precision when as many keyphrase matchings have been retrieved as gold standard keyphrases are assigned to the document. An R-precision of 1.0 is equivalent to perfect keyphrase ranking and perfect recall.

These properties make R-p a favorable metric for keyphrase extraction, as it puts a focus on the precision at the first ranks, which is necessary for most practical systems that assign or present only a handful of keyphrases. R-p also measures whether the keyphrases at the first ranks cover the whole set of topics in the document. For example, a keyphrase extraction approach that extracts a lot of variants (e.g. "scheduling", "real-time scheduling", "embedded real-time scheduling") at the first ranks will have a lower precision than an approach that covers more topics. As an additional benefit, R-p is a single number metric allowing for more compact presentation of results and easier comparison.

We define R-p using the following formalization: R-p is defined as the precision

		Judges accepting matchings	
	#	4	≥ 3
LONGER	274	.58	.80
SHORTER	239	.31	.44
MORPH	53	.96	.96
MORPH+LONGER	327	.65	.83

Table 7.2: Ratio of approximate keyphrase matchings acceptable to human judges (4 = all judges; ≥ 3 = at least 3 out of 4 judges).

when $|M| = |K_{gold}|$. Precision is computed as $\frac{|M_c|}{|M|}$, where M_c is the list of correct matchings and M is the full list of matchings.

Approximate Matching Strategy The exact matching strategy EXACT is only partially indicative of the performance of a keyphrase extraction method, as it is known to underestimate performance as perceived by human judges (Turney, 2000). Additionally, it may not be a good indicator of the overall quality of the extracted set of keyphrases, as there are many cases in which exact matching fails, e.g. lexical-semantic variations (*automobile sales*, *car sales*), overlapping phrases (*scheduling*, *real-time scheduling*), or morphological variants like plurals (*performance metric*, *performance metrics*).[4] Thus, we propose a new approximate matching strategy APPROX(k_{gold}, k_{ext}) that accounts for morphological variants (MORPH) and the two cases of overlapping phrases: either the extracted keyphrase is longer and includes the gold standard keyphrase (LONGER) or the extracted keyphrase is shorter and a part of the gold standard keyphrase (SHORTER). We leave the inclusion of (i) lexical-semantic variations and (ii) other morphological variations to future work. Exact matchings are of course still valid in addition to approximate matchings.

For overlapping phrases, we do not allow character level variations, but only token level variations, i.e. the LONGER category contains matchings where the extracted keyphrase contains all the tokens in the gold keyphrase plus some additional tokens. For the SHORTER matchings, it works the other way round. In the case of the morphological variants MORPH, we limit approximate matching to the detection of plurals.

Approximate Matching Evaluation For testing whether the new approximate matching strategy is acceptable to humans, we randomly selected a maximum of 300 non-exact matchings from each of the three datasets (giving a maximum of 900 randomly selected matchings). We included matchings from each of the 3 approximate matching categories (LONGER, SHORTER, and MORPH) using different candidate selection methods and length restrictions to account for all kinds of keyphrase variants. The total number of selected approximate matchings is 566, as some matchings were included in multiple sets of the random matchings and morphological approximate matching MORPH did not yield 100 approximate matchings per dataset.

Four judges annotated whether it would be acceptable to replace the gold standard keyphrase with the extracted keyphrase using the approximate matching strat-

[4]In the remainder of this section, we present example matchings as (*gold keyphrase*, *extracted keyphrase*).

egy. As no context was given when judging about a matching, annotators where instructed to annotate a pair as invalid if in doubt. Thus, the annotation has a pessimistic bias and rather underestimates human agreement with the approximate matching. The results of the annotation study are presented in Table 7.2.

In the MORPH category of morphological variants, agreement between judges was very high: 96% of all MORPH matchings were acceptable to all 4 judges. The only problematic case were two abbreviations (*fms*, *fmss*) and (*soa*, *soas*) where the judges could not decide about the validity without looking at the context. Agreement between all 4 judges is considerably lower for the LONGER and SHORTER categories. However, given the inherent subjectivity of the task, we treat an agreement of 3 out of 4 judges as valid for accepting a match. In the LONGER category agreement reaches 80%, while for the SHORTER category agreement is only 44%.

The major source of error in the LONGER category was wrong preprocessing. For example, the matching (*security level*, *give security level*) was unanimously rejected by all judges, as the extracted keyphrase contains a chunking error. A major source of error in the SHORTER category were cases when the extracted keyphrase is too general compared to the gold keyphrase, e.g. (*topic importance*, *topic*). A potential refinement of the SHORTER heuristic would be to match only extracted keyphrases whose head noun matches the head of the gold keyphrase. However, only 52% of such cases (66 out of 128) were accepted by at least 3 judges. Furthermore, in 35% of the cases (39 out of 111) a matching with a non-matching head like (*tuberculosis cases*, *tuberculosis*) was accepted by at least 3 judges. This means, neither is a matching head required for a keyphrase to be acceptable to human judges, nor is a matching head sufficient for an acceptable match. As we aim at high precision approximate matching, we decided not to use the SHORTER category due to these problems, but combined MORPH and LONGER to an approximate matching strategy with a human agreement of 83%. The new strategy is better suited to assess the quality of extracted keyphrases as perceived by humans. The approximate matching strategy is formally defined as: $\text{APPROX}(k_{gold}, k_{ext}) = \text{EXACT} \vee \text{MORPH} \vee \text{LONGER}$.

Limitations and Future Work

In future research, we want to improve the approximate matching strategy, as it currently does not address lexical-semantic variations as well as more complicated morphological variations. Also, for languages other than English with higher morphological variability or free word order, the methods for finding overlapping keyphrases used in this thesis might not be sufficient. In the future, we also want to further investigate under which circumstances extracted keyphrases that partially match the annotated gold standard keyphrases are acceptable to humans. Another research direction is to include paraphrase recognition in the evaluation process, as matchings like (*topic importance*, *importance of a topic*) are currently not covered.

7.1.5 Experimental Results

For our comprehensive analysis, we set aside two thirds of the documents in each dataset for training, while the rest of the data is used for evaluation. Note that all keyphrase extraction methods except *Kea* did not make use of the training data.

		Inspec		DUC		SP	
	Candidates	R-p$_{ex}$	R-p$_{ap}$	R-p$_{ex}$	R-p$_{ap}$	R-p$_{ex}$	R-p$_{ap}$
KEA	N-grams	.16	.19	.11	.14	**.21**	**.25**
tf.idf	Tokens (N,A)	.27	.32	.12	.15	.12	.22
TextRank	Tokens (N,A)	**.31**	**.36**	**.21**	**.23**	.04	.10
	Tokens	.11	.22	.05	.12	.06	.18
	Tokens (N,A)	.27	**.32**	**.12**	.15	.12	**.22**
	Lemmas	.15	.27	.06	.14	.07	.21
tf.idf	Lemmas (N,A)	**.28**	**.32**	**.12**	**.16**	**.13**	**.22**
	N-grams	.10	.16	.03	.06	.06	.15
	Noun Phrases	.27	.32	**.12**	.14	.10	.21
	Named Entities	.01	.01	.11	.13	.06	.08
	Tokens	.06	.22	.00	.07	.00	.05
	Tokens (N,A)	**.31**	**.36**	.21	.23	.04	.10
	Lemmas	.07	.22	.00	.06	.00	.06
co-occ	Lemmas (N,A)	.29	.35	**.22**	**.24**	.08	.15
	N-grams	.07	.22	.03	.10	.01	.09
	Noun Phrases	.28	.34	.12	.14	**.12**	**.18**
	Named Entities	.01	.01	.09	.09	.04	.05

Table 7.3: State-of-the-art keyphrase extraction results in terms of R-precision using exact matching (R-p$_{ex}$) and approximate matching (R-p$_{ap}$). Best values for each section are in bold.

However, as we wanted to ensure a fair comparison, we tested all keyphrase extraction systems on the same evaluation data.

We selected three reference systems: tf.idf ranking as a simple baseline, *Kea* (Frank et al., 1999) as the most widely used supervised system, and *TextRank* (Mihalcea and Tarau, 2004) as the state-of-the-art unsupervised system. All reference systems are included into our generalized framework for keyphrase extraction (as shown in Figure 7.3) to ensure a fair comparison.

We first discuss results of state-of-the-art approaches on the three evaluation datasets. As our keyphrase extraction framework allows to use a wider range of candidate selection methods than used in previous work, we already test system configurations that go beyond the state of the art. We then select the best performing configurations and compare them to our LSG approach.

State-of-the-art approaches We compare the three reference systems *Kea*, standard tf.idf, and *TextRank* with all possible combinations of the candidate selection strategies and the ranking methods (i) tf.idf ranking and (ii) co-occurrence graph based ranking (abbreviated as "co-occ"). For comparison of the exact matching and the approximate matching strategy, we computed both R-precision for exact matching (R-p$_{ex}$) and approximate matching (R-p$_{ap}$). Table 7.3 gives an overview of the obtained results.[5]

Theoretically, *Kea* as a supervised system is expected to yield the best performance. Tf.idf ranking based methods (that do not use any training data, but use information drawn from the whole document collection) are supposed to perform

[5]Note that in our framework, the *TextRank* system is equivalent to using Token (N,A) as the candidate selection strategy and using co-occurrence graph based ranking. We duplicated this row of results as 'TextRank' for convenience.

worse than supervised systems, but better than co-occurrence graph based methods like *TextRank* that only use information from a single document. However, under the controlled conditions of our keyphrase extraction framework, the unsupervised *TextRank* outperforms *Kea* by a wide margin on the Inspec and on the DUC dataset. Both datasets contain only rather small documents ($\approx$ 100–1000 tokens), making it relatively easy to select the correct keyphrases.

On the SP dataset containing the longer documents, *Kea* outperforms all co-occurrence or tf.idf based system configurations by a wide margin when using exact matching. However, the approximate matching strategy reveals that the performance gap between the best configuration using tf.idf ranking with *Lemma (N,A)* candidates and *Kea* is not as large as exact matching indicates (dropping from .08 to .03).

The wide range of candidates tested within our framework allows to draw other interesting conclusions: The candidate selection strategies *Tokens*, *Lemmas*, and *N-grams* generally lead to poor performance due to the over-generation of candidates. In most cases, *Lemma (N,A)* candidates perform slightly better than *Tokens (N,A)* candidates, but the small difference does not justify the additional effort of lemmatization. The *TextRank* result on the SP dataset can almost be doubled (from .10 to .18 R-p_{ap}) by using noun phrases instead of Tokens (N,A) as candidates. This indicates that using higher quality candidates can have a positive impact on keyphrase extraction performance on longer documents.

Lexical-Semantic Graphs First, we performed additional experiments beyond the scope of this discussion to determine the best candidates to be used with lexical-semantic graphs. We found that *Noun Phrases* candidates always outperformed the other candidate selection approaches when used with LSG based ranking. Thus, we only report results based on *Noun Phrases* candidates. We also do not use path based measures in our experiments, due to performance reasons.

Tables 7.4 (a) and (b) give an overview of the obtained results using gloss based and vector based measures. Among the gloss based measures, WordNet and Wiktionary pseudo glosses perform best. On the Inspec and the DUC datasets containing rather short documents, all resources perform comparably. On the SP dataset containing longer documents, WordNet and Wiktionary outperform Wikipedia by a wide margin. For the vector based measures, we find that all resources, except using the full Wikipedia for creating concept vectors, yield comparable performance.

Table 7.5 summarizes our results showing the best gloss based and vector based results compared to the best configurations of state-of-the-art approaches. On the Inspec dataset, the gloss based LSG and the vector based LSG show equal performance and are among the best performing measures. On the DUC dataset, both LSG approaches do not reach the performance level of the co-occurrence based methods, but still outperform the supervised *Kea* and the tf.idf approach. Vector based LSG slightly outperforms gloss based LSG, which is in contrast to the SP dataset where it is the other way round. On the SP dataset, the gloss based LSG approach shows a remarkable performance that outperforms all co-occurrence and tf.idf based approaches, and almost reaches the performance of the supervised *Kea* approach. However, this is only true when looking at the approximate matching results (*Kea* = .25; LSG Gloss-best = .24). For exact matching, *Kea* (.21) still outperforms LSG

(a) Gloss based LSG results.

		Inspec		DUC		SP	
Resource	Gloss type	R-p$_{ex}$	R-p$_{ap}$	R-p$_{ex}$	R-p$_{ap}$	R-p$_{ex}$	R-p$_{ap}$
WordNet	normal	**.30**	.36	.10	.12	.07	.09
	pseudo	**.30**	**.37**	**.12**	**.15**	**.14**	**.24**
Wiktionary	normal	**.30**	**.37**	.10	.12	.07	.10
	pseudo	**.30**	**.37**	**.12**	**.15**	**.14**	**.24**
Wikipedia	normal	.28	.34	.09	.11	.07	.09
	first	.29	.35	.11	.14	.10	.15

(b) Vector based LSG results.

	Inspec		DUC		SP	
Resource	R-p$_{ex}$	R-p$_{ap}$	R-p$_{ex}$	R-p$_{ap}$	R-p$_{ex}$	R-p$_{ap}$
WordNet	**.30**	.36	.15	**.19**	.08	.19
Wiktionary	**.30**	**.37**	**.16**	**.19**	**.09**	**.20**
Wikipedia	.29	.36	.12	.15	.08	.15
Wikipedia-first	**.30**	**.37**	.15	**.19**	**.09**	**.20**

Table 7.4: Keyphrase extraction results using gloss and vector based LSG in terms of R-precision using exact matching (R-p$_{ex}$) and approximate matching (R-p$_{ap}$). Best values are in bold.

		Inspec		DUC		SP	
	Candidates	R-p$_{ex}$	R-p$_{ap}$	R-p$_{ex}$	R-p$_{ap}$	R-p$_{ex}$	R-p$_{ap}$
KEA	N-grams	.16	.19	.11	.14	**.21**	**.25**
TextRank	Tokens (N,A)	**.31**	.36	.21	.23	.04	.10
tf.idf	Tokens (N,A)	.27	.32	.12	.15	.12	.22
tf.idf	Tokens (N,A)	.27	.32	.12	.15	.12	.22
	Lemmas (N,A)	.28	.32	.12	.16	.13	.22
	Noun Phrases	.27	.32	.12	.14	.10	.21
co-occ	Tokens (N,A)	**.31**	.36	.21	.23	.04	.10
	Lemmas (N,A)	.29	.35	**.22**	**.24**	.08	.15
	Noun Phrases	.28	.34	.12	.14	.12	.18
LSG	Gloss-best	.30	**.37**	.12	.15	.14	.24
	Vector-best	.30	**.37**	.16	.19	.09	.20

Table 7.5: Comparison of keyphrase extraction results in terms of R-precision using exact matching (R-p$_{ex}$) and approximate matching (R-p$_{ap}$). Best values are in bold.

Gloss-best (.14) by a wide margin. On the SP dataset in general, we observe quite high performance increases between R-p_{ex} and R-p_{ap} for the unsupervised tf.idf, co-occurrence, and LSG based approaches, while the performance increase for the supervised *Kea* system is rather small. Thus, we conclude that getting the exact keyphrase boundaries seems a quite complicated task on longer documents. While R-p_{ap} reveals that most approaches find almost as many correct keyphrases as *Kea*, we could not have come to that conclusion when only looking at the results from exact matching.

7.1.6 Summary

In this section, we introduced the task of keyphrase extraction. We presented a new evaluation strategy for keyphrase extraction based on approximate keyphrase matching that accounts for the shortcomings of exact matching. In an annotation study, we showed that approximate matching (based on morphological variants and extracted keyphrases including the gold standard keyphrases) corresponds well with human judgments. We showed that the approximate matching strategy is better suited to assess the performance of keyphrase extraction approaches.

We developed a new keyphrase extraction approach based on semantic relatedness measures with the goal to find infrequently used words in a document that are semantically connected to many other words in the document. We proposed a generalized framework for the comprehensive analysis and evaluation of keyphrase extraction systems, and integrated our new approach into the framework. We also integrated the state-of-the-art systems *Kea* and *TextRank*, as well as a baseline system based on tf.idf ranking.

Our experimental results show that for small and medium sized documents ($\approx$ 100–1000 tokens), the unsupervised approaches outperform the supervised system by a wide margin. Gloss and vector based LSG perform comparably to the other systems. On larger documents, the supervised system outperforms all other approaches, but using approximate matching reveals that the gloss based LSG approach is the best unsupervised approach that almost reaches the performance of the supervised *Kea* system.

7.2 Other applications

In this section, we describe other applications in which semantic relatedness measures of this thesis have been used to improve the results of simple string based methods.

7.2.1 Semantic Information Retrieval

Gurevych et al. (2007) describe work on semantic information retrieval in the domain of electronic career guidance. The task of electronic career guidance is to support school leavers in their search for a profession or a vocational training. Vocational trainings are represented by documents which were automatically extracted from BERUFEnet, a database created by the German Federal Labour Office. The

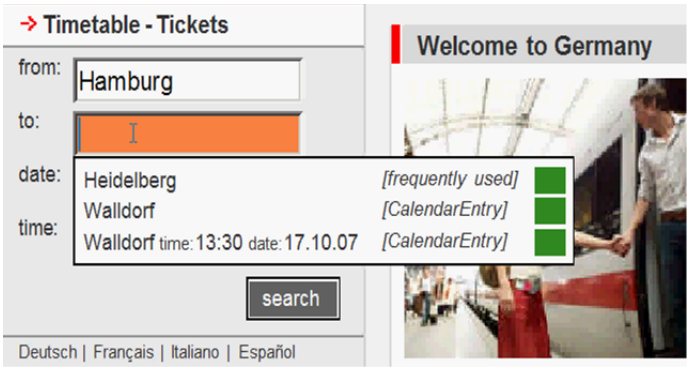


Figure 7.7: Example of a context-aware user interface.

queries are short essays collected from students in which they describe in their own words what they would like their future job to be like. One special challenge of this task is the large *vocabulary gap* between the language of the (expert-authored) documents from the database and the language of the students. The term 'vocabulary gap' relates to the fact that people with different backgrounds or different levels of expertise use (sometimes strikingly) different vocabularies when describing similar things. String-based information retrieval approaches (as represented e.g. by the standard information retrieval system *Lucene*) are not able to adequately handle this phenomenon. The best results reported by Gurevych et al. (2007) are yielded by a semantic information retrieval model using the IC-based Lin98 measure or the vector based GM07 measure.

Müller et al. (2008a) further investigate the integration of semantic relatedness measures into information retrieval algorithms. They find that using semantic relatedness gives consistently better results, especially for shorter queries, because of semantic relatedness measures bridging the vocabulary gap between query and document. They also find that the information about very strongly related terms is most beneficial for semantic information retrieval.

Müller and Gurevych (2009) combine Wikipedia and Wiktionary as semantic resources for a concept vector based measure, which increases the results of information retrieval. In a second bilingual evaluation, a mapping between concepts in semantic resources is used to retrieve documents in a language not matching the query language. The approach yields significant performance increases.

To summarize, semantic relatedness measures are able to bridge the vocabulary gap inherent to most information retrieval tasks. Using the collaboratively constructed semantic resources Wikipedia and Wiktionary yields performance increases in monolingual retrieval and at the same time enables multilingual retrieval due to Wikipedia's and Wiktionary's availability for many languages.

7.2.2 Context-Aware User Interfaces

Hartmann et al. (2008) describe work on context-aware user interfaces that use semantic relatedness measures to derive suggestions for input elements in Web applications. Context-aware user interfaces target the problem that the increasing amount of features available in today's applications often leads to a decreased us-

ability of the user interface. Context-aware user interfaces counter this problem by facilitating the user interaction by suggesting or prefilling data derived from the user's current context. Figure 7.7 shows an example context-aware user interface that suggests possible destinations on a train booking website from the user's context (e.g. from her calendar or frequently used items). A main problem in this setting is how to decide which context objects relate to a certain input element. For example, the user's calendar (a possible source of context) might label the destination of a travel as "Destination", while the input element might be labelled "Travel to". Hartmann et al. (2008) address this problem for Web applications by automatically extracting a textual representation of the website's input elements. They then compute the semantic relatedness between these textual representations and the context information to bridge possible vocabulary gaps.

Experimental results show that within a certain domain (e.g. 'car booking' or 'hotel booking'), semantic relatedness measures do not improve the results over a substring match baseline (Hartmann et al., 2008), as the vocabulary gap is small. Results are different when trying to match across domains, e.g. between "pick-up date" from the 'car booking' domain and "check-in date" from the 'hotel booking' domain (Hartmann and Mühlhäuser, 2009). Depending on the domains, using semantic relatedness measures slightly improves matching results over a substring matching baseline.

7.3 Chapter Summary

In this chapter, we described three real-world applications in which semantic relatedness measures from this thesis have been put to use. When applied to *keyphrase extraction*, semantic relatedness based approaches performe comparably to other unsupervised approaches on short documents (100–1000 tokens). On larger documents, they outperformed the other unsupervised approaches and performed comparably to a supervised state-of-the-art system. Semantic relatedness measures have also shown to improve the application performance over string based approaches in the context of *semantic information retrieval* and *context-aware user interfaces*.

Chapter 8

Summary

In this thesis, we conducted a comprehensive study of semantic relatedness. We showed that the collaboratively constructed resources Wikipedia and Wiktionary are promising semantic resources containing a rich variety of lexical-semantic information. We gave a comprehensive overview of the state of the art in computing semantic relatedness using different kinds of semantic resources. We categorized existing semantic relatedness measures into four types which make use of different properties of a semantic resource. We investigated how each of these measure types can be adapted to all semantic resources considered in this thesis. We verified the validity of this adaptation process by a graph-theoretic analysis showing that all semantic resource graph structures have similar properties.

We tested the adaptation of semantic relatedness measures from linguistically to collaboratively constructed semantic resources. For that purpose, we used two intrinsic evaluation tasks: (i) comparison with human judgments, and (ii) solving word choice problems. For the first task, we found that vector based measures yield the best correlation with human judgments on non-classical semantic relationships between words. For the second task of solving word choice problems, we found that the coverage of a semantic resource is crucial for the final performance. Collaboratively constructed resources either have the same coverage as classical resources (English) or have a superior coverage (German).

We also performed an extrinsic application based evaluation using semantic relatedness measures for keyphrase extraction. We developed a new keyphrase extraction approach based on semantic relatedness measures. The new approach is designed to find infrequently used words in a document that are semantically connected to many other words in the document and carry its essential meaning. We developed a generalized framework for the comprehensive analysis and evaluation of keyphrase extraction systems. Our experimental results show that for small and medium sized documents ($\approx$ 100–1000 tokens), all unsupervised approaches (including our new approach) outperform the supervised system by a wide margin. On larger documents, the supervised system outperforms all other approaches, but our new approach almost reaches the performance of the supervised system.

In order to conduct this extrinsic evaluation, we developed a new evaluation strategy based on approximate keyphrase matching. In an annotation study, we showed that the new evaluation strategy corresponds well with human judgments, and is better suited to assess the performance of keyphrase extraction approaches

as compared to previously used evaluation strategies.

Overall, we found that collaboratively constructed semantic resources can fully substitute linguistically constructed semantic resources for the task of computing semantic relatedness. When compared to large linguistically constructed semantic resources (e.g. WordNet for English), they yield comparable results, while for languages like German, with less developed linguistically constructed semantic resources, using collaboratively constructed semantic resources even leads to performance increases. Vector based semantic relatedness measures have shown to be the most versatile measure type showing good performance and being easily applicable to all semantic resources. We also show (on the example of the German Wikipedia) that the growth of a resource has no or little negative effect on the performance of semantic relatedness measures, but that the coverage steadily increases. This makes the resources more useful in the context of large-scale natural language processing applications, where coverage is a main criterion for overall performance. Thus, collaboratively constructed semantic resources can indeed be used as a proxy for linguistically created semantic resources that might not exist for minor languages.

Future Work

In future work, we want to include more collaboratively constructed semantic resources into our evaluation framework. Additionally, distributional semantic relatedness measures should be compared to the knowledge based measures targeted in this thesis. As intrinsic evaluation is currently limited by the size of the evaluation datasets, larger datasets should be created. Evaluation datasets should also cover a wider range of languages. In this thesis, we focused on generalizing semantic relatedness measures in a way that makes them applicable to a wide range of semantic resources. Thus, we did not fully explore the space of possible adaptations of semantic relatedness measures to the peculiarities of certain semantic resources. It would also be interesting to combine largely complementary semantic resources like Wikipedia and Wiktionary.

One of our future research directions should be to investigate whether the growth of collaboratively constructed semantic resources has an influence on extrinsic evaluation tasks like keyphrase extraction. With respect to keyphrase extraction, we want to further refine our lexical-semantic graph approach, and to improve the approximate matching strategy, as it currently does not address lexical-semantic variations as well as more complex morphological variations. Also, for languages other than English with higher morphological variability or free word order, the methods for finding overlapping keyphrases used in this thesis might not be sufficient. It is also necessary to further investigate under which circumstances extracted keyphrases that partially match the annotated gold standard keyphrases are acceptable to humans. Another research direction is to include paraphrase recognition into the evaluation process, as matchings like (*topic importance*, *importance of a topic*) are currently not covered.

Bibliography

Ahn, D., Jijkoun, V., Mishne, G., Müller, K., de Rijke, M., and Schlobach, S. (2004). Using Wikipedia at the TREC QA Track. In *Proceedings of the Thirteenth Text REtrieval Conference (TREC)*, Gaithersburg, Maryland.

Anscombe, F. J. (1973). Graphs in statistical analysis. *American Statistician*, 27:17–21.

Banerjee, S. and Pedersen, T. (2002). An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet. In *CICLing '02: Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing*, pages 136–145, London, UK. Springer-Verlag.

Barabasi, A.-L. and Albert, R. (1999). Emergence of Scaling in Random Networks. *Science*, 286:509–512.

Barker, K. and Cornacchia, N. (2000). Using Noun Phrase Heads to Extract Document Keyphrases. In *Canadian Conference on AI*, pages 40–52, Montréal, Quebec, Canada. Springer.

Bauer, C. and King, G. (2004). *Hibernate in Action. Practical Object/Relational Mapping.* Manning.

Berrey, L. and Carruth, G., editors (1962). *Roget's international thesaurus.* Thomas Y. Crowell Co., New York, third edition edition.

Biemann, C. (2005). Semantic Indexing with Typed Terms Using Rapid Annotation. In *Proceedings of the TKE-05-Workshop on Methods and Applications of Semantic Indexing*, Copenhagen.

Bouchard, A., Liang, P., Griffiths, T., and Klein, D. (2007). A Probabilistic Approach to Diachronic Phonology. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 887–896, Prague, Czech Republic.

Boyd-Graber, J., Fellbaum, C., Osherson, D., and Shapire, R. (2006). Adding Dense, Weighted, Connections to WordNet. In *Proceedings of the Third Global WordNet Meeting*, Jeju Island, Korea.

Bracewell, D. B., Ren, F., and Kuriowa, S. (2005). Multilingual Single Document Keyword Extraction for Information Retrieval. In *Proceedings of 2005 IEEE International Conference on Natural Language Processing and Knowledge Engineering*, pages 517–522, Wuhan, China.

Budanitsky, A. and Hirst, G. (2006). Evaluating WordNet-based Measures of Semantic Distance. *Computational Linguistics*, 32(1):13–47.

Bunescu, R. and Pasca, M. (2006). Using Encyclopedic Knowledge for Named Entity Disambiguation. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 9–16, Trento, Italy.

Buriol, L., Castillo, C., Donato, D., Leonardi, S., and Millozzi, S. (2006). Temporal Analysis of the Wikigraph. In *Proceedings of Web Intelligence*, pages 45–51, Hong Kong.

Capocci, A., Servedio, V. D. P., Colaiori, F., Buriol, L. S., Donato, D., Leonardi, S., and Caldarelli, G. (2006). Preferential Attachment in the Growth of Social Networks: The Internet Encyclopedia Wikipedia. *Physical Review E*, 74:036116.

Chesley, P., Vincent, B., Xu, L., and Srihari, R. (2006). Using Verbs and Adjectives to Automatically Classify Blog Sentiment. In *Proceedings of AAAI-CAAW-06, the Spring Symposia on Computational Approaches to Analyzing Weblogs*, pages 27–29, Stanford University, California, USA.

Cilibrasi, R. L. and Vitanyi, P. M. (2007). The Google Similarity Distance. *IEEE Transactions on Knowledge and Data Engineering*, 19(3):370–383.

Csomai, A. and Mihalcea, R. (2007). Investigations in Unsupervised Back-of-the-Book Indexing. In *Proceedings of the Florida Artificial Intelligence Research Society (FLAIRS 2007)*, pages 211–216, Key West.

Culo, O., Kunz, K., and Zesch, T. (2009). Semantic Relations in a Bilingual Corpus of Different Registers. In *Deutsche Gesellschaft für Sprachwissenschaft (DGfS) Workshop on Corpus, Colligation, Register Variation*, Osnabrück.

D'Avanzo, E. and Magnini, B. (2005). A Keyphrase-Based Approach to Summarization: the LAKE System at DUC-2005. In *Proceedings of DUC Workshop at HLT/EMNLP'05*, Vancouver, B.C., Canada.

Eckart de Castilho, R. and Gurevych, I. (2009). DKPro-UGD: A Flexible Data-Cleansing Approach to Processing User-Generated Discourse. In *Online-proceedings of the First French-speaking Meeting around the Framework Apache UIMA*, Nantes, France. LINA CNRS UMR 6241 - University of Nantes.

Fellbaum, C. (1990). English Verbs as a Semantic Net. *International Journal of Lexicography*, 3(4):278–301.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA.

Ferrucci, D. and Lally, A. (2004). UIMA: An Architectural Approach to Unstructured Information Processing in the Corporate Research Environment. *Natural Language Engineering*, 10(3-4):327–348.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 363–370, Ann Arbor, Michigan.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., and Wolfman, G. (2002). Placing Search in Context: The Concept Revisited. *ACM Transactions on Information Systems*, 20(1):116–131.

Frank, E., Paynter, G. W., Witten, I., Gutwin, C., and Nevill-Manning, C. G. (1999). Domain-Specific Keyphrase Extraction. In *Proceedings of the 16th International Joint Conference on Aritificial Intelligence (IJCAI)*, pages 668–673, San Mateo, CA. Morgan Kaufmann.

Fröhner, T., Nickles, M., Weiß, G., Brauer, W., and Franken, R. (2005). Integration of Ontologies and Knowledge from Distributed Autonomous Sources. *Künstliche Intelligenz*, pages 18–23.

Gabrilovich, E. and Markovitch, S. (2006). Overcoming the Brittleness Bottleneck using Wikipedia: Enhancing Text Categorization with Encyclopedic Knowledge. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference*, pages 1301–1306, Boston, Massachusetts, USA. AAAI Press.

Gabrilovich, E. and Markovitch, S. (2007). Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. In *Proceedings of The 20th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1606–1611, Hyderabad, India.

Galley, M. and McKeown, K. (2003). Improving Word Sense Disambiguation in Lexical Chaining. In *Proceedings of 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, pages 1486–1488, Acapulco, Mexico.

Garoufi, K., Zesch, T., and Gurevych, I. (2008a). Graph-Theoretic Analysis of Collaborative Knowledge Bases in Natural Language Processing. In *Proceedings of the Poster Session of the 7th International Semantic Web Conference (ISWC)*, Karlsruhe, Germany.

Garoufi, K., Zesch, T., and Gurevych, I. (2008b). Representational Interoperability of Linguistic and Collaborative Knowledge Bases. In *Proceedings of the KONVENS Workshop on Lexical-Semantic and Ontological Resources – Maintenance, Representation, and Standards*, Berlin, Germany.

Giles, J. (2005). Internet encyclopaedias go head to head. *Nature*, 438(7070):900–901.

Gross, D. and Miller, K. J. (1990). Adjectives in WordNet. *International Journal of Lexicography*, 3(4):265–277.

Gurevych, I. (2005). Using the Structure of a Conceptual Network in Computing Semantic Relatedness. In *Proceedings of the 2nd International Joint Conference on Natural Language Processing*, pages 767–778, Jeju Island, Republic of Korea.

Gurevych, I. (2006). Computing Semantic Relatedness Across Parts of Speech. Technical report, Darmstadt University of Technology, Germany, Department of Computer Science, Telecooperation.

Gurevych, I., Müller, C., and Zesch, T. (2007). What to be? - Electronic Career Guidance Based on Semantic Relatedness. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 1032–1039, Prague, Czech Republic.

Gurevych, I. and Strube, M. (2004). Semantic Similarity Applied to Spoken Dialogue Summarization. In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING)*, pages 764–770, Geneva, Switzerland.

Gutwin, C., Paynter, G., Witten, I., Nevill-Manning, C., and Frank, E. (1999). Improving Browsing in Digital Libraries with Keyphrase Indexes. *Decision Support Systems*, 27(1-2):81–104.

Halliday, M. A. K. and Hasan, R. (1976). *Cohesion in English*. Longman, London.

Hammouda, K. M., Matute, D. N., and Kamel, M. S. (2005). CorePhrase: Keyphrase Extraction for Document Clustering. *Machine Learning and Data Mining in Pattern Recognition*, 2005:265–274.

Hartmann, M. and Mühlhäuser, M. (2009). Context-Aware Form Filling for Web Applications. In *Proceedings of the Third IEEE International Conference on Semantic Computing*, Berkeley, CA, USA.

Hartmann, M., Zesch, T., Mühlhäuser, M., and Gurevych, I. (2008). Using Similarity Measures for Context-Aware User Interfaces. In *Proceedings of the 2nd IEEE International Conference on Semantic Computing*, pages 190–197, Santa Clara, CA, USA.

Hirst, G. and St-Onge, D. (1998). *WordNet: An Electronic Lexical Database*, chapter Lexical Chains as Representation of Context for the Detection and Correction of Malapropisms, pages 305–332. Cambridge: MIT Press.

Hulth, A. (2003). Improved Automatic Keyword Extraction Given More Linguistic Knowledge. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'03)*, pages 216–223, Sapporo.

Hulth, A. (2004). Enhancing Linguistically Oriented Automatic Keyword Extraction. In *Proceedings of the Human Language Technology Conference/North American Chapter of the Association for Computational Linguistics Annual Meeting: Short Papers*, pages 17–20, Boston, Massachusetts, USA.

Jacobi, C. (2007). Using Wikipedia in the Example Application Domain "Electronic Career Guidance". (Orig. title "Nutzung von Wikipedia am Fallbeispiel Elektronische Berufsberatung"). Master's thesis, Computer Science Department. Technische Universität Darmstadt.

Jarmasz, M. and Szpakowicz, S. (2003). Roget's Thesaurus and Semantic Similarity. In *Proceedings of Recent Advances in Natural Language Processing (RANLP)*, pages 111–120, Borovets, Bulgaria.

Jeh, G. and Widom, J. (2002). SimRank: A Measure of Structural-context Similarity. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 538–543, Edmonton, Alberta, Canada.

Jiang, J. J. and Conrath, D. W. (1997). Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In *Proceedings of the 10th International Conference on Research in Computational Linguistics*, Taipei, Taiwan.

Kaplan, A. N. and Schubert, L. K. (2001). Measuring and Improving the Quality of World Knowledge Extracted from WordNet. Tech. Rep. 751 14627-0226, Dept. of Computer Science, Univ. of Rochester, Rochester, NY.

Kluck, M. (2004). The GIRT Data in the Evaluation of CLIR Systems - from 1997 Until 2003. *Lecture Notes in Computer Science*, 3237:376–390.

Kulessa, S. (2008). Mining Wikipedia's Revision History for Paraphrase Extraction. Master's thesis, Computer Science Department. Technische Universität Darmstadt.

Kunze, C. (2004). *Lexikalisch-semantische Wortnetze*, chapter Computerlinguistik und Sprachtechnologie, pages 423–431. Spektrum Akademischer Verlag.

Leacock, C. and Chodorow, M. (1998). *WordNet: An Electronic Lexical Database*, chapter Combining Local Context and WordNet Similarity for Word Sense Identification, pages 265–283. Cambridge: MIT Press.

Lemnitzer, L. and Kunze, C. (2002). GermaNet - Representation, Visualization, Application. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC)*, pages 1485–1491.

Lenat, D. and Guha, R. (1990). *Building Large Knowledge-Based Systems: Representation and Inference in the Cyc Project*. Addison Wesley Publishing Company.

Lesk, M. (1986). Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the 5th Annual International Conference on Systems Documentation*, pages 24–26, Toronto, Ontario, Canada.

Li, Y., Bandar, Z. A., and McLean, D. (2003). An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources. *IEEE Transactions on Knowledge and Data Engineering*, 15(4):871–882.

Lin, D. (1998). An Information-Theoretic Definition of Similarity. In *Proceedings of International Conference on Machine Learning*, pages 296–304, Madison, Wisconsin.

Litvak, M. and Last, M. (2008). Graph-Based Keyword Extraction for Single-Document Summarization. In *Proceedings of the 24nd International Conference on Computational Linguistics (COLING)*, pages 17–24.

Matsuo, Y. and Ishizuka, M. (2004). Keyword Extraction from a Single Document Using Word Co-occurrence Statistical Information. *International Journal on Artificial Intelligence Tools*, 13(1):157–169.

McHale, M. (1998). A Comparison of WordNet and Roget's Taxonomy for Measuring Semantic Similarity. *Computing Research Repository (CoRR)*, cmp-lg/9809003.

Medelyan, O. and Legg, C. (2008). Integrating Cyc and Wikipedia: Folksonomy meets rigorously defined common-sense. In *Proceedings of the WikiAI Workshop at AAAI-2008*, pages 13–18.

Medelyan, O. and Witten, I. H. (2006). Thesaurus Based Automatic Keyphrase Indexing. In *In Proceedings of the Joint Conference on Digital Libraries (JCDL)*, pages 296–297, Chapel Hill, NC, USA.

Medelyan, O., Witten, I. H., and Milne, D. (2008). Topic Indexing with Wikipedia. In *Proceeding of AAAI Workshop on Wikipedia and Artificial Intelligence: An Evolving Synergy*, pages 19–24, Chicago, USA. AAAI Press.

Mihalcea, R. and Moldovan, D. (2001). Automatic Generation of a Coarse Grained WordNet. In *Proceedings of NAACL Workshop on WordNet and Other Lexical Resources*, Pittsburgh, PA.

Mihalcea, R. and Moldovan, D. I. (1999). A Method for Word Sense Disambiguation of Unrestricted Text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 152–158, College Park, Maryland, USA.

Mihalcea, R. and Tarau, P. (2004). TextRank: Bringing Order into Texts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, pages 404–411, Barcelona, Spain.

Miller, G. A. (1990). Nouns in WordNet: A Lexical Inheritance System. *International Journal of Lexicography*, 3(4):245–264.

Miller, G. A. and Charles, W. G. (1991). Contextual Correlates of Semantic Similarity. *Language and Cognitive Processes*, 6(1):1–28.

Milne, D. (2007). Computing Semantic Relatedness using Wikipedia Link Structure. In *Proceedings of the New Zealand Computer Science Research Student Conference (NZCSRSC 2007)*, Hamilton, New Zealand.

Milne, D. and Witten, I. H. (2008a). An Effective, Low-Cost Measure of Semantic Relatedness Obtained from Wikipedia Links. In *Proceedings of the first AAAI Workshop on Wikipedia and Artificial Intelligence (WIKIAI'08)*, pages 25–30, Chicago, USA.

Milne, D. and Witten, I. H. (2008b). Learning to Link With Wikipedia. In *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM)*, pages 509–518, Napa Valley, California.

Milne, D. and Witten, I. H. (2009). An Open-Source Toolkit for Mining Wikipedia. In *Online Proceedings of the New Zealand Computer Science Research Student Conference*, Auckland, New Zealand.

Mohammad, S., Gurevych, I., Hirst, G., and Zesch, T. (2007). Cross-lingual Distributional Profiles of Concepts for Measuring Semantic Distance. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 571–580, Prague, Czech Republic.

Morris, J. and Hirst, G. (1991). Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text. *Computational Linguistics*, 17(1):21–48.

Morris, J. and Hirst, G. (2004). Non-Classical Lexical Semantic Relations. In *Workshop on Computational Lexical Semantics, Human Language Technology Conference of the North American Chapter of the ACL*, pages 46–51, Boston.

Müller, C. and Gurevych, I. (2009). Using Wikipedia and Wiktionary in Domain-Specific Information Retrieval. In Peters, C., Giampiccol, D., Ferro, N., Petras, V., Gonzalo, J., Penas, A., Deselaers, T., Mandl, T., Jones, G., and Kurimo, M., editors, *Evaluating Systems for Multilingual and Multimodal Information Access – 9th Workshop of the Cross-Language Evaluation Forum*, Lecture Notes in Computer Science. Springer-Verlag GmbH, Berlin / Heidelberg.

Müller, C., Gurevych, I., and Mühlhäuser, M. (2008a). Closing the Vocabulary Gap for Computing Text Similarity and Information Retrieval. *International Journal of Semantic Computing*, 2(2):253–272. World Scientific Publishing Company.

Müller, C., Zesch, T., Müller, M.-C., Bernhard, D., Ignatova, K., Gurevych, I., and Mühlhäuser, M. (2008b). Flexible UIMA Components for Information Retrieval Research. In *Proceedings of the LREC 2008 Workshop 'Towards Enhanced Interoperability for Large HLT Systems: UIMA for NLP'*, pages 24–27, Marrakech, Morocco.

Nakayama, K., Hara, T., and Nishio, S. (2007). Wikipedia Mining for an Association Web Thesaurus Construction. In *Proceedings of International Conference on Web Information Systems Engineering (WISE)*, pages 322–334, Nancy, France.

Nelson, D. L., McEvoy, C. L., and Schreiber, T. A. (1998). The University of South Florida Word Association, Rhyme, and Word Fragment Norms. Technical report, University of South Florida, `http://www.usf.edu/FreeAssociation/`.

Newman, M. E. J. (2003). The Structure and Function of Complex Networks. *SIAM Review*, 45:167–256.

Nguyen, T. D. and Kan, M.-Y. (2007). Keyphrase Extraction in Scientific Publications. In *Proceedings of International Conference on Asian Digital Libraries*, pages 317–326, Hanoi, Vietnam.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The PageRank Citation Ranking: Bringing Order to the Web. Technical Report 1999-66, Stanford InfoLab.

Park, Y., Byrd, R. J., and Boguraev, B. K. (2002). Automatic Glossary Extraction: Beyond Terminology Identification. In *Proceedings of the 19th International Conference on Computational Linguistics*, pages 1–7, Morristown, NJ, USA.

Patwardhan, S., Banerjee, S., and Pedersen, T. (2003). Using Measures of Semantic Relatedness for Word Sense Disambiguation. In *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*, pages 241–257, Mexico City.

Patwardhan, S. and Pedersen, T. (2006). Using WordNet Based Context Vectors to Estimate the Semantic Relatedness of Concepts. In *Proceedings of the EACL 2006 Workshop Making Sense of Sense - Bringing Computational Linguistics and Psycholinguistics Together*, pages 1–8, Trento, Italy.

Paukkeri, M.-S., Nieminen, I. T., Pöllä, M., and Honkela, T. (2008). A Language-Independent Approach to Keyphrase Extraction and Evaluation. In *Proceedings of the 24nd International Conference on Computational Linguistics (COLING) Companion volume: Posters*, pages 83–86, Manchester, UK.

Pirró, G. and Seco, N. (2008). Design, Implementation and Evaluation of a New Semantic Similarity Metric Combining Features and Intrinsic Information Content. In *OTM '08: Proceedings of the OTM 2008 Confederated International Conferences, CoopIS, DOA, GADA, IS, and ODBASE 2008. Part II on On the Move to Meaningful Internet Systems*, pages 1271–1288, Berlin, Heidelberg. Springer-Verlag.

Procter, P. (1978). *Longman Dictionary of Contemporary English*. Longman.

Qiu, Y. and Frei, H. (1993). Concept Based Query Expansion. In *Proceedings of the 16th ACM International Conference on Research and Development in Information Retrieval*, pages 160–169, Pittsburgh, PA, USA.

Rada, R., Mili, H., Bicknell, E., and Blettner, M. (1989). Development and Application of a Metric on Semantic Nets. *IEEE Trans. on Systems, Man, and Cybernetics,*, 19(1):17–30.

Resnik, P. (1995). Using Information Content to Evaluate Semantic Similarity. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 448–453, Montreal, Canada.

Riddle, T. (2006). Parse::MediaWikiDump. URL `http://search.cpan.org/~triddle/Parse-MediaWikiDump-0.40/`.

Rubenstein, H. and Goodenough, J. B. (1965). Contextual Correlates of Synonymy. *Communications of the ACM*, 8(10):627–633.

Ruiz-Casado, M., Alfonseca, E., and Castells, P. (2005). Automatic Assignment of Wikipedia Encyclopedic Entries to WordNet Synsets. In *Advances in Web Intelligence*, pages 380–386.

Salton, G. (1989). *Automatic Text Processing: the Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Salton, G. and McGill, M. J. (1983). *Introduction to Modern Information Retrieval*. McGraw-Hill, New York.

Schmid, H. (1995). Probabilistic Part-of-Speech Tagging Using Decision Trees. In *International Conference on New Methods in Language Processing*, Manchester, UK.

Schulte im Walde, S. and Melinger, A. (2005). Identifying Semantic Relations and Functional Properties of Human Verb Associations. In *Proceedings of the joint Conference on Human Language Technology and Empirical Methods in NLP*, pages 612–619, Vancouver, Canada.

Seco, N. (2005). Computational Models of Similarity in Lexical Ontologies. Master's thesis, University College Dubin.

Seco, N. and Hayes, T. V. J. (2004). An Intrinsic Information Content Metric for Semantic Similarity in WordNet. In *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI)*, Valencia, Spain.

Shanks, B. (2005). WikiGateway: A Library for Interoperability and Accelerated Wiki Development. In *WikiSym '05: Proceedings of the 2005 International Symposium on Wikis*, pages 53–66, New York, NY, USA. ACM Press.

Shi, L. and Mihalcea, R. (2005). Putting Pieces Together: Combining FrameNet, VerbNet and WordNet for Robust Semantic Parsing. In *Proceedings of the Sixth International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*, pages 100–111, Mexico City, Mexico. Springer.

Silber, H. G. and McCoy, K. F. (2002). Efficiently Computed Lexical Chains as an Intermediate Representation for Automatic Text Summarization. *Computational Linguistics*, 28(4):487–496.

Stevenson, M. and Greenwood, M. A. (2005). A Semantic Approach to IE Pattern Induction. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 379–386, Morristown, NJ, USA.

Steyvers, M. and Tenenbaum, J. B. (2005). The Large-Scale Structure of Semantic Networks: Statistical Analyses and a Model of Semantic Growth. *Cognitive Science*, 29:41–78.

Strube, M. and Ponzetto, S. (2006). WikiRelate! Computing Semantic Relatedness Using Wikipedia. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI-06)*, pages 1419–1424, Boston, Mass.

Suchanek, F. M., Kasneci, G., and Weikum, G. (2007). Yago: A Core of Semantic Knowledge. In *Proceedings of the 16th International World Wide Web Conference (WWW)*, pages 697–706, New York, NY, USA. ACM Press.

Summers, E. (2006). WWW:Wikipedia. URL `http://search.cpan.org/~esummers/WWW-Wikipedia-1.9/`.

Tomokiyo, T. and Hurst, M. (2003). A Language Model Approach to Keyphrase Extraction. In *Proceedings of the ACL 2003 Workshop on Multiword Expressions*, pages 33–40, Morristown, NJ, USA.

Turney, P. (2006). Expressing Implicit Semantic Relations without Supervision. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the ACL*, pages 313–320, Sydney, Australia.

Turney, P. D. (2000). Learning Algorithms for Keyphrase Extraction. *Information Retrieval*, 2:303–336.

Turney, P. D. (2003). Coherent Keyphrase Extraction via Web Mining. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 434–439, Acapulco, Mexico.

Völkel, M., Krötzsch, M., Vrandecic, D., Haller, H., and Studer, R. (2006). Semantic Wikipedia. In *Proceedings of the 15th International Conference on World Wide Web*, pages 585–594, Edinburgh, Scotland.

Voss, J. (2005). Measuring Wikipedia. In *Proceedings of the 10th International Conference of the International Society for Scientometrics and Informetrics (ISSI)*, Stockholm, Sweden.

Voss, J. (2006). Collaborative Thesaurus Tagging the Wikipedia Way. *The Computing Research Repository (CoRR)*, abs/cs/0604036.

Vossen, P. (1998). Introduction to EuroWordNet. *Computers and the Humanities. Special Issue on EuroWordNet.*, 32(2–3):73–89.

Wallace, D. and Wallace, L. A. (2001–2005). *Reader's Digest, das Beste für Deutschland*. Jan 2001–Dec 2005. Verlag Das Beste, Stuttgart.

Wan, X. and Xiao, J. (2008). Single Document Keyphrase Extraction Using Neighborhood Knowledge. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*, pages 855–860, Chicago, USA.

Watts, D. J. and Strogatz, S. H. (1998). Collective Dynamics of Small-World Networks. *Nature*, 393:440–442.

Weeds, J. and Weir, D. (2005). Co-occurrence Retrieval: A Flexible Framework for Lexical Distributional Similarity. *Computational Linguistics*, 31(4):439–475.

Wilks, Y., Fass, D., Guo, C.-M., McDonald, J., Plate, T., and Slator, B. (1990). Providing Machine Tractable Dictionary Tools. *Journal of Machine Translation*, 5(2):99–151.

Wu, Z. and Palmer, M. (1994). Verb Semantics and Lexical Selection. In *32nd Annual Meeting of the ACL*, pages 133–138, New Mexico State University, Las Cruces, New Mexico.

Yang, D. and Powers, D. M. W. (2006). Verb Similarity on the Taxonomy of WordNet. In *Proceedings of the Third International WordNet Conference (GWC-06)*, pages 121–128, Jeju Island, Korea.

Zesch, T. and Gurevych, I. (2006). Automatically Creating Datasets for Measures of Semantic Relatedness. In *Proceedings of the COLING/ACL Workshop on Linguistic Distances*, pages 16–24, Sydney, Australia.

Zesch, T. and Gurevych, I. (2007). Analysis of the Wikipedia Category Graph for NLP Applications. In *Proceedings of the TextGraphs-2 Workshop (NAACL-HLT 2007)*, pages 1–8, Rochester, NY, USA.

Zesch, T. and Gurevych, I. (2009). Approximate Matching for Evaluating Keyphrase Extraction. In *Proceedings of the 7th International Conference on Recent Advances in Natural Language Processing*, pages 484–489, Borovets, Bulgaria.

Zesch, T. and Gurevych, I. (2010). Wisdom of Crowds versus Wisdom of Linguists - Measuring Relatedness of Words. *Journal of Natural Language Engineering*, 16.

Zesch, T., Gurevych, I., and Mühlhäuser, M. (2007a). Analyzing and Accessing Wikipedia as a Lexical Semantic Resource. In Rehm, G., Witt, A., and Lemnitzer, L., editors, *Data Structures for Linguistic Resources and Applications*, pages 197–205. Gunter Narr, Tübingen, Tuebingen, Germany.

Zesch, T., Gurevych, I., and Mühlhäuser, M. (2007b). Comparing Wikipedia and German Wordnet by Evaluating Semantic Relatedness on Multiple Datasets. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2007)*, pages 205–208, Rochester, NY, USA.

Zesch, T., Müller, C., and Gurevych, I. (2008a). Extracting Lexical Semantic Knowledge from Wikipedia and Wiktionary. In *Proceedings of the Conference on Language Resources and Evaluation (LREC)*, Marrakech, Morocco. electronic proceedings.

Zesch, T., Müller, C., and Gurevych, I. (2008b). Using Wiktionary for Computing Semantic Relatedness. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*, pages 861–867, Chicago, IL, USA.

Zlatic, V., Bozicevic, M., Stefancic, H., and Domazet, M. (2006). Wikipedias: Collaborative web-based encyclopedias as complex networks. *Physical Review E*, 74:016115.

Appendix A

Enabling Technologies

The experiments presented in this thesis required a set of specific enabling technologies that were not publicly available before. For that reason, we developed (i) the Wikipedia application programming interface *JWPL* (described in Section A.1), and (ii) a system for semi-automatically creating datasets for the evaluation of semantic relatedness measures called *DEXTRACT* (described in Section A.2). Additionally, we augmented the UIMA software component repository DKPro (described in Section A.3) to enable the experiments performed in this thesis. We also implemented (i) a representational interoperability framework for semantic resources called *Lexical-Semantic Resource Interface* that was already described in Section 2.4, and (ii) a generalized framework for keyphrase extraction that was already described in Section 7.1.

A.1 JWPL

Linguistically constructed semantic resources like WordNet (Fellbaum, 1998) or GermaNet (Kunze, 2004) are usually shipped with easy-to-use application programming interfaces (APIs), e.g. JWNL[1] or GermaNetAPI[2], that allow for easy integration into applications. However, Wikipedia has lacked this kind of support so far which constitutes a significant impediment for NLP research. Therefore, we developed a general purpose, high performance Java-based API for Wikipedia called JWPL. In this thesis, it was used for the graph-theoretic analysis of the Wikipedia category graph in Section 4.1.3, and for all the experiments involving Wikipedia described in Chapter 6 and Chapter 7. JWPL is publicly available for research purposes from `http://www.ukp.tu-darmstadt.de/software/jwpl`.

JWPL operates on an optimized database that is created from the database dumps available from the Wikimedia foundation.[3] The structure of the database corresponding to these dumps is optimized for searching articles by keywords which is performed by millions of users of the online Wikipedia every day. However, an API designed for NLP research has to support a wider range of access paths (including iteration over all articles), a query syntax, as well as efficient access to sources

[1] `http://sourceforge.net/projects/jwordnet`

[2] `http://projects.villa-bosch.de/nlpsoft/gn_api/index.html`

[3] `http://download.wikipedia.org/`

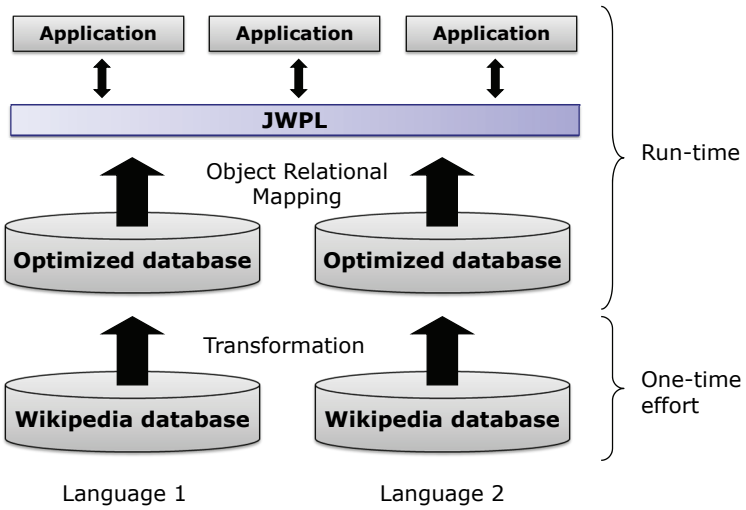


Figure A.1: System architecture of JWPL.

of lexical-semantic information like links, categories, and redirects. Thus, JWPL operates on an optimized database (as shown in Figure A.1) that is created in a one-time effort from the original dumps. In the optimized database, lexical-semantic information (e.g. redirects) is explicitly stored with the corresponding article making them easily accessible.

JWPL accesses the database using object-relational mapping (ORM). ORM bridges the impedance mismatch between relational databases and object-oriented programming languages. The impedance mismatch occurs because relational databases store data as "rows and columns", while object-oriented programming relies on complex objects. Thus, we cannot read these objects directly from the database. At this point, ORM can be applied. A mapping file tells the relational database how a complex object should be mapped to a relational database scheme. Thus, objects can be read from the database in a transparent manner. ORM even transparently updates the database, when the object is changed in the Java program. This guarantees a high stability and maintainability of the Wikipedia API. Additionally, ORM abstracts further from the actual database structure and, thus, fully decouples the API design from a particular database or a particular underlying database scheme. The *advantages* of the described system architecture are:

Computational efficiency enables large-scale NLP tasks like computing semantic relatedness. Computational efficiency is also a consequence of accessing the database using its indexing mechanisms for fast retrieval. The data from the database is directly mapped to Java objects using the Hibernate object-relational mapping framework (Bauer and King, 2004). This also means that JWPL is not restricted to using a certain database, but may run on top of the most common database systems.[4]

Reproducible research results Reproducible experimental results are a direct consequence of using a fixed database dump instead of the online Wikipedia that is very likely to change between two runs of a certain experimental setting.

[4] `http://www.hibernate.org/80.html`

Easy to use object-oriented programming interface The design of the programming interface is centered around the objects: WIKIPEDIA, PAGE, and CATEGORY. The WIKIPEDIA object is used to establish the connection with the database, and to retrieve PAGE and CATEGORY objects. JWPL supports retrieval by keywords or via a query interface that allows for wildcard matches as well as retrieving subsets of articles or categories depending on parameters like the number of tokens in an article or the number of ingoing links. The WIKIPEDIA object also allows to iterate over articles, categories, redirects, and disambiguation pages.

A PAGE object represents either a normal Wikipedia article, a redirect to an article, or a disambiguation page. Each PAGE object provides access to the article text (with markup information or as plain text), the assigned categories, the ingoing and outgoing article links, as well as all redirects that link to this article.

CATEGORY objects represent Wikipedia categories and allow access to the articles within this category. As categories in Wikipedia form a thesaurus, a CATEGORY object also provides means to retrieve parent and child categories, as well as siblings and all recursively collected descendants. JWPL also provides a CATEGORYGRAPH object that e.g. allows to find the shortest path between two given categories (as shown in Listing A.4). Section A.1 contains Java code, which exemplifies the use of the API for some basic tasks.

Language independence JWPL is applicable to all Wikipedia language editions, and abstracts over their peculiarities. For example, the top-most category is called *Categories* in English Wikipedia, while it is *!Hauptkategorie* in German Wikipedia. The transformation step maps all language specific features into a generalized representation.

Advanced Functionality JWPL also contains a parser for the Wikipedia markup language (Jacobi, 2007). The parser allows to easily identify and access even more fine-grained information within Wikipedia articles, e.g. sections, paragraphs, templates, links, link texts, link anchors, lists, and tables. Figure A.2 visualizes the structure of the Wikipedia article "Natural Language Processing" as analyzed by the parser.

A recent addition to JWPL (Kulessa, 2008) enables access to the revision history of Wikipedia articles. This allows for (i) reconstructing a certain state of Wikipedia in the past, and (ii) analyzing the collaborative writing process that led to the current state of an article. The reconstruction of past Wikipedia states was of crucial importance for the analysis of the Wikipedia growth in Chapter 6.

Java code examples

Listing A.1: Getting the article page with title "Benedikt XVI" and accessing page text and redirects.

```
Wikipedia wiki = new Wikipedia();
Page page = wiki.getPage("Benedikt XVI");
```

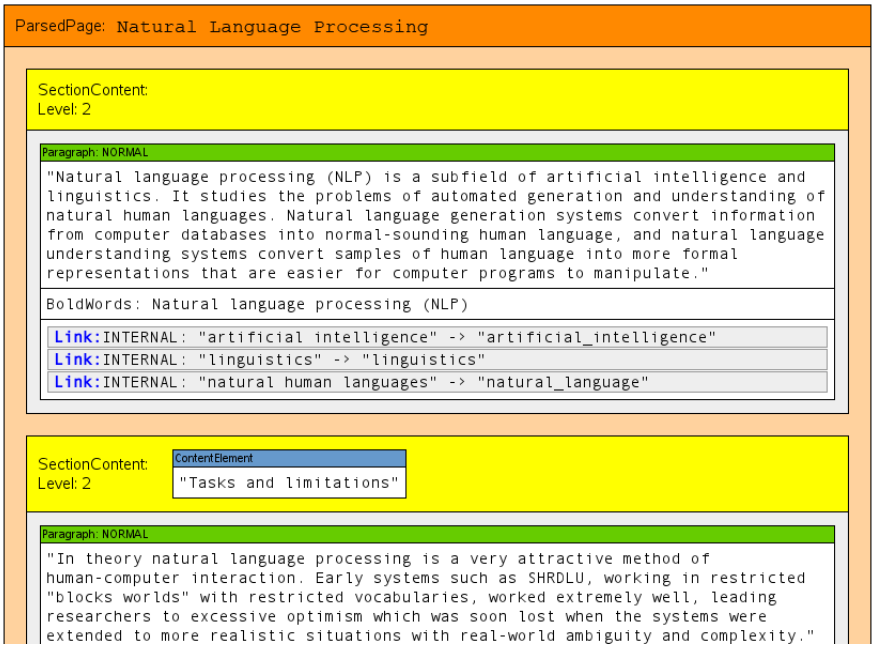


Figure A.2: Visualization of the structure of a Wikipedia article as analyzed by the parser.

```
String pageText = page.getText();
Set<String> redirects = page.getRedirects();
```

Listing A.2: Getting all Wikipedia pages except disambiguation pages having at least 5 redirects, and containing at least 100 words.

```
Wikipedia wiki = new Wikipedia();
Query query = new Query();
query.setMinRedirects(5);
query.setMinTokens(100);
query.setDisambiguationPages(false);
Set<Integer> pageIds = wiki.getPages(query);
```

Listing A.3: Getting a list of all Wikipedia article titles about *Wirbeltiere* (Eng. *vertebrates*), i.e. a list of vertebrates.

```
Wikipedia wiki = new Wikipedia();
Category cat = wiki.getCategory("Wirbeltiere");
Set<Category> subCats = cat.getSubCategories(wiki);

List<Integer> pageIds = new ArrayList<Integer>();
for (Category subCat : subCats) {
    pageIds.addAll(subCat.getPages());
}

List<String> mammals = new ArrayList<String>();
for (int pageId : pageIds) {
    Page p = wiki.getPage(pageId);
```

```
        if (!p.isDisambiguation()) {
            mammals.add(p.getName());
        }
    }
```

Listing A.4: Getting the path length between the categories *England* and *Deutschland*.

```
Wikipedia wiki = new Wikipedia();
CategoryGraph cg = new CategoryGraph();
Category c1 = wiki.getCategory("England");
Category c2 = wiki.getCategory("Deutschland");
int pathLength = cg.getPathLengthInNodes(
                    c1.getPageId(),c2.getPageId()
                );
```

Comparison with Other Approaches

The simplest way to retrieve a Wikipedia page is to enter a search term on the Wikipedia Web site.[5] However, this approach is not suited for automatic access to Wikipedia articles by an application. The Perl module WWW::Wikipedia (Summers, 2006) offers simple means for retrieving Wikipedia pages by programmatically querying the Wikipedia Web site. However, this approach poses enormous load on the Wikipedia servers when used in large-scale applications. Therefore, it is discouraged by the Wikimedia Foundation.

Other approaches relying on Web crawling and thus also not being suited for large-scale NLP applications are: (i) the Wikipedia bot framework (available for different programming languages like Python[6] or Java[7]) that can be used to create small programs called *bots* acting on behalf of a normal user and usually employed for maintenance tasks, (ii) the Wiki Gateway tool box, a unified API for interfacing with a variety of remote wiki engines (Shanks, 2005), and (iii) the system developed by Strube and Ponzetto (2006) relying on a modified version of the WWW::Wikipedia module to retrieve articles.

Crawling can be avoided by running an own server using publicly available Wikipedia database dumps.[8] This gives better, but still insufficient performance, due to the overhead related to using a Web server for delivering the retrieved pages. In this setting, retrieving a Wikipedia article usually involves a transfer of the request from an application to the Web server. The Web server then executes a PHP script that accesses the Wikipedia database, and the database returns the article content encoded using Wiki markup[9] to the PHP script which converts the Wiki markup to HTML. Finally, the Web server delivers the HTML encoded data back to the application. This poses a substantial overhead that might render large-scale NLP tasks impossible.

[5] http://www.wikipedia.org/
[6] http://pywikipediabot.sourceforge.net/
[7] http://jwbf.sourceforge.net/
[8] http://download.wikipedia.org/
[9] http://en.wikipedia.org/wiki/WP:MARKUP

This overhead can be avoided by directly accessing the database dumps. For example, the Perl module Parse::MediaWikiDump (Riddle, 2006) parses the Wikipedia XML dump to retrieve articles. As Wikipedia dumps are very large (over 3 GB of compressed data for the snapshot of the English Wikipedia from Feb 2008), the performance of parsing is not sufficient for large-scale NLP tasks (it may take up to several seconds to retrieve a given article). Additionally, the time that is required to retrieve an article is not easily predictable, but depends on the article's position in the XML dump.

WikiPrep (Gabrilovich and Markovitch, 2007) is a preprocessor that transforms a Wikipedia XML dump into an optimized XML format that explicitly encodes information such as the category hierarchy or article redirects. However, as the resulting data is still in XML format, WikiPrep suffers from the same performance problem as Parse::MediaWikiDump.

WikipediaMiner (Milne and Witten, 2009) is a recently released open-source tool written in Perl and Java that offers a similar functionality as JWPL.

A.2 DEXTRACT

DEXTRACT (Zesch and Gurevych, 2006) is a tool, implementing a semi-automatic corpus-based approach for creating evaluation datasets for semantic relatedness measures. DEXTRACT takes a corpus as input, and outputs an evaluation dataset containing a list of automatically generated word pairs. As these word pairs are selected automatically, they cannot be biased towards strong classical relations beyond corpus evidence as it is the case with word pairs selected by humans that tend to select only highly related pairs connected by relations they are aware of (Gurevych, 2006). However, randomly generating word pairs from the corpus would result in too many unrelated pairs. Thus, words are assigned to word pairs according to their tf.idf weights. Then, a set of user defined filters is applied, normally including a stopword filter removing stopwords, and a part-of-speech based filter that forces the final evaluation dataset to contain a specified number of word pairs with certain part-of-speech combinations. DEXTRACT is publicly available for research purposes from `http://www.ukp.tu-darmstadt.de/software/dextract`.

A.2.1 System architecture

Figure A.3 gives an overview of the system architecture of DEXTRACT. In the first step, a source corpus is preprocessed using tokenization, part-of-speech tagging and lemmatization resulting in a list of part-of-speech tagged lemmas. Randomly generating word pairs from this list would result in too many unrelated pairs, yielding an unbalanced dataset. Thus, we assign weights to each word (e.g. using tf.idf-weighting). The most important document-specific words are assigned the highest weights and due to lexical cohesion of the documents many related words can be found among the top rated. Therefore, we randomly generate a user-defined number of word pairs from the words with the highest weights for each document.

In the next step, user defined filters are applied to the initial list of word pairs. For example, a filter can remove all pairs containing only uppercase letters (mostly

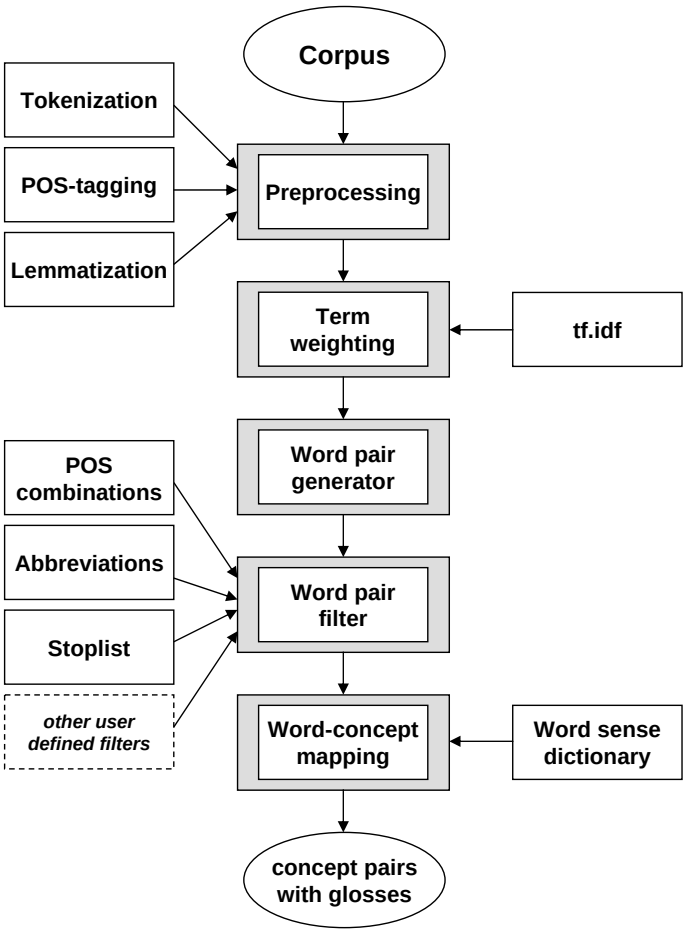


Figure A.3: System architecture for extracting concept pairs.

acronyms). Another filter can enforce a certain fraction of part-of-speech combinations to be present in the result set.

As we want to obtain judgment scores for semantic relatedness of concepts instead of words, we have to include all word sense combinations of a pair in the list. An external dictionary of word senses is necessary for this step. It is also used to add a gloss for each word sense that enables test subjects to distinguish between senses.

If differences in meaning between senses are very fine-grained, distinguishing between them is hard even for humans (Mihalcea and Moldovan, 2001). For example, the German verb "halten" (possible English translations are "hold", "maintain", "present", "sustain", etc. already indicating its ambiguity) has 26 senses in GermaNet. Pairs containing such words are not suitable for evaluation. To limit their impact on the experiment, a threshold for the maximal number of senses can be defined. Words with a number of senses above the threshold are removed from the list. The result of the extraction process is a list of sense disambiguated, part-of-speech tagged pairs of concepts.

A.2.2 Experimental setup

We extracted word pairs from three different domain-specific corpora (see Table A.1):

Corpus	# Docs	# Tokens	Domain
BN	9,022	7,728,501	descriptions of professions
GIRT	151,319	19,645,417	abstracts of social science papers
SPP	106	144,074	scientific .ppt presentations

Table A.1: Corpus statistics.

- The *BERUFEnet* (BN) corpus[10] consists of descriptions of 5,800 professions in Germany and therefore contains many terms specific to professional training. Evaluating semantic relatedness on an evaluation dataset based on this corpus may reveal the ability of a measure to adapt to a very special domain.
- The *GIRT* (German Indexing and Retrieval Testdatabase) corpus (Kluck, 2004) is a collection of abstracts of social science papers. It is a standard corpus for evaluating German information retrieval systems.
- The third corpus is compiled from 106 arbitrarily selected *scientific PowerPoint presentations* (SPP). They cover a wide range of topics from bio genetics to computer science and contain many technical terms. Due to the special structure of presentations, this corpus will be particularly demanding with respect to the required preprocessing components of an information retrieval system.

The three preprocessing steps (tokenization, part-of-speech tagging, lemmatization) are performed using TreeTagger (Schmid, 1995). The resulting list of part-of-speech tagged lemmas is weighted using the SMART 'ltc'[11] tf.idf-weighting scheme (Salton, 1989).

In the resulting list of word pairs, we remove a word pair if it contains at least one word that a) has less than three letters b) contains only uppercase letters (mostly acronyms), or c) can be found in a stoplist. Another filter enforces a specified fraction of combinations of nouns (N), verbs (V) and adjectives (A) to be present in the result set. We used the following parameters: $NN = 0.5$, $NV = 0.15$, $NA = 0.15$, $VV = 0.1$, $VA = 0.05$, $AA = 0.05$. That means 50% of the resulting word pairs for each corpus were noun-noun pairs, 15% noun-verb pairs and so on.

Word pairs containing polysemous words are expanded to concept pairs using GermaNet (Kunze, 2004) as a sense inventory for each word. However, GermaNet contains only a few conceptual glosses that are required to enable test subjects to distinguish between senses. Thus, we use artificial glosses composed from synonyms and hypernyms as a surrogate, e.g. for *brother*: "brother, male sibling" vs. "brother, comrade, friend" (Gurevych, 2005). We removed words which had more than three senses. Finally, marginal manual post-processing was necessary, since the lemmatization process introduced some errors. Foreign words were translated into German, unless they are common technical terminology.

[10] `http://berufenet.arbeitsagentur.de`

[11] l=logarithmic term frequency, t=logarithmic inverse document frequency, c=cosine normalization.

Figure A.4: Screenshot of the DEXTRACT graphical user interface. Polysemous words are defined by means of synonyms and related words.

We initially selected 100 word pairs from each corpus. 11 word pairs were removed because they comprised non-words. Expanding the word list to a concept list increased the size of the list. Thus, the final dataset contained 328 automatically created concept pairs.

Graphical User Interface We developed a Web based interface to obtain human judgments of semantic relatedness for each automatically generated concept pair (see Figure A.4). Test subjects were invited via email to participate in the experiment, i.e. they were not supervised during the experiment.

Gurevych (2006) observed that some annotators were not familiar with the exact definition of semantic relatedness. Their results differed particularly in cases of ANTONYMY or distributional related pairs. We created a manual with a detailed introduction to semantic relatedness stressing the crucial points. The manual was presented to the subjects before the experiment and could be re-accessed at any time. During the experiment, one concept pair at a time was presented to the test subjects in random order. Subjects had to assign a discrete relatedness value from the range {0,1,2,3,4} to each pair.

In case of a polysemous word, synonyms or related words were presented to enable test subjects to understand the sense of a presented concept. As this additional information can lead to undesirable priming effects, test subjects were instructed to deliberately decide only about the relatedness of a concept pair and use the gloss solely to understand the sense of the presented concept. Since our corpus-based approach includes domain-specific vocabulary, we could not assume that the subjects

	Concepts		Words	
	Inter	**Intra**	**Inter**	**Intra**
all	.48	.65	.49	.68
BN	.47	.70	.50	.72
GIRT	.45	.60	.46	.63
SPP	.54	.65	.52	.68
AA	.56	.89	.60	.89
NA	.55	.77	.51	.76
NV	.51	.66	.54	.65
NN	.46	.62	.48	.66
VA	.32	.32	.39	.21
VV	.28	.49	.30	.48

Table A.2: Inter-annotator agreement grouped by corpus and PoS combinations.

were familiar with all words. Thus, they were instructed to look up unknown words in the German Wikipedia.[12]

Several test subjects were asked to repeat the experiment with a minimum break of one day. Results from the repetition can be used to measure intra-subject correlation. They can also be used to obtain some hints on varying difficulty of judgment for special concept pairs or parts-of-speech.

A.2.3 Results and discussion

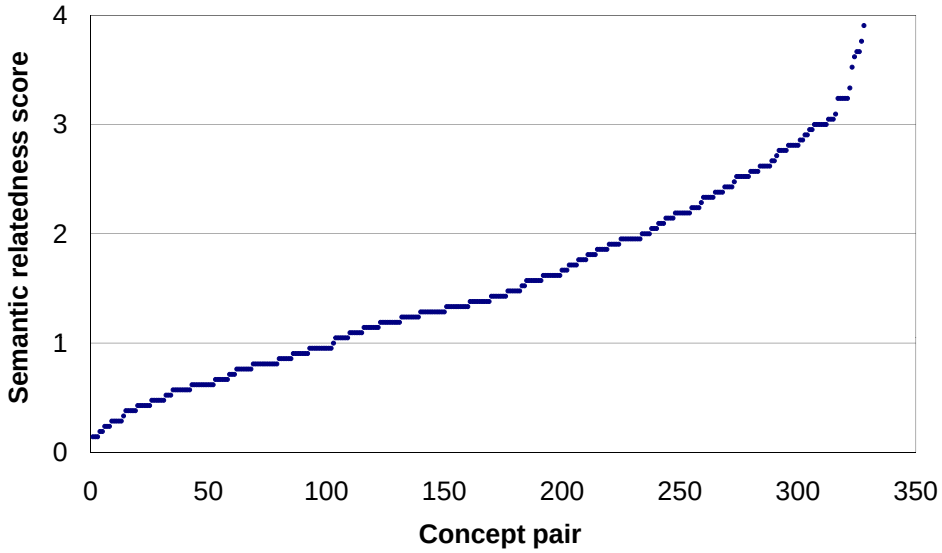
21 test subjects (13 males, 8 females) participated in the experiment, two of them repeated it. The average age of the subjects was 26 years. Most subjects had an computer science background. The experiment took 39 minutes on average, i.e. it took a subject about 7 seconds for rating a concept pair.

The inter-annotator agreement between the 21 subjects was $r = .48$ (cf. Table A.2), which is statistically significant at $p < .05$. This correlation coefficient is an upper bound of performance for automatic SR measures applied on the same dataset. Compared with other studies, the correlation is rather low (cf. Table 5.2).

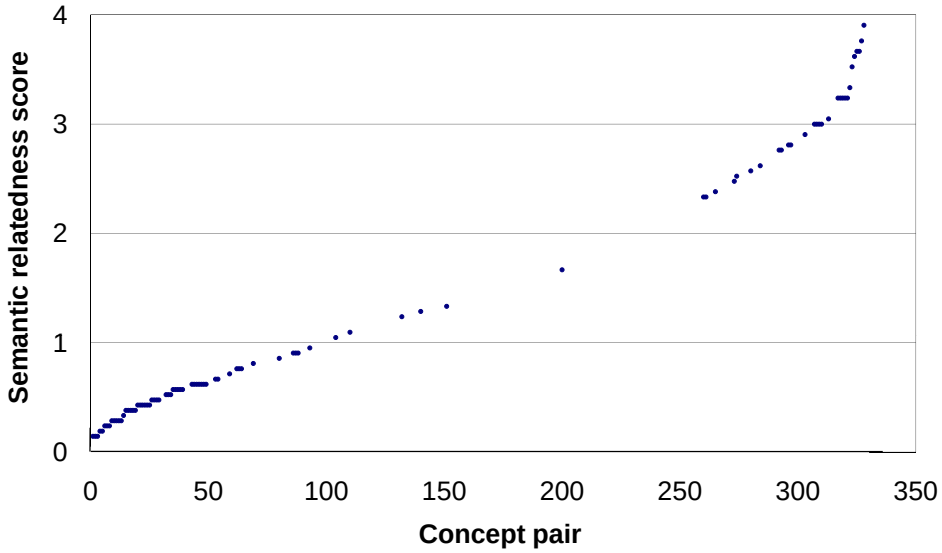
Evaluating the influence of using concept pairs instead of word pairs is complicated because word level judgments are not directly available. Therefore, we computed a lower and an upper bound for correlation coefficients. For the lower bound, we always selected the concept pair with the highest standard deviation from each set of corresponding concept pairs. The upper bound is computed by selecting the concept pair with the lowest standard deviation. The differences between correlation coefficient for concepts and words are not significant. Table A.2 shows only the lower bounds.

Correlation coefficients for experiments measuring semantic relatedness are expected to be lower than results for semantic similarity, since the former also includes additional relations (like co-occurrence of words) and is thus a more complex task. Judgments for such relations strongly depend on experience and cultural background of the test subjects. While most people may agree that (*car* – *vehicle*) are highly related, a strong connection between (*parts* – *speech*) may only be recognized by a certain group (e.g. computational linguists). Due to the corpus-based

[12] `http://www.wikipedia.de`



(a) All judgments.



(b) Judgments with standard deviation ≤ 0.9.

Figure A.5: Distribution of averaged human judgments.

approach, many domain-specific concept pairs are introduced into the evaluation dataset. Therefore, inter-subject correlation is lower than the results obtained by Gurevych (2006).

In our experiment, intra-subject correlation was r=.670 for the first and r=.623 for the second individual who repeated the experiment, yielding a summarized intra-subject correlation of r=.647. Rubenstein and Goodenough (1965) reported an intra-subject correlation of r=.85 for 15 subjects judging the similarity of a subset (36) of the original 65 word pairs. The values may again not be compared directly. Furthermore, we cannot generalize from these results, because the number of participants which repeated our experiment was too low.

The distribution of averaged human judgments on the whole evaluation dataset

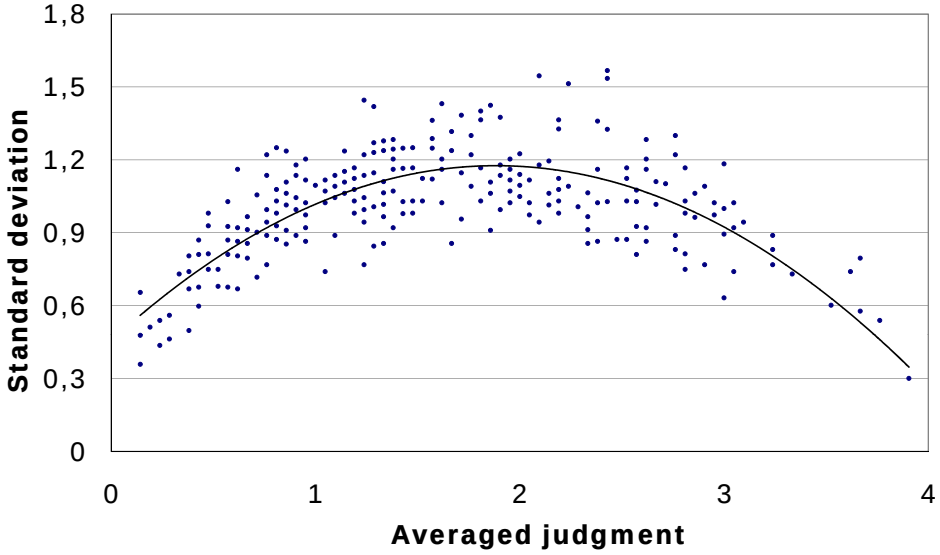


Figure A.6: Averaged judgments and standard deviation for all concept pairs. Low deviations are only observed for low or high judgments.

(see Figure A.5(a)) is almost balanced with a slight under-representation of highly related concepts. To create more highly related concept pairs, more sophisticated weighting schemes or selection on the basis of lexical chaining could be used. However, even with the present setup, automatic extraction of concept pairs performs remarkably well and can be used to quickly create balanced evaluation datasets.

Budanitsky and Hirst (2006) pointed out that distribution plots of judgments for the word pairs used by Rubenstein and Goodenough display an empty horizontal band that could be used to separate related and unrelated pairs. This empty band is not observed here. However, Figure A.5(b) shows the distribution of averaged judgments with the highest agreement between annotators (standard deviation ≤ 0.9). The plot clearly shows an empty horizontal band with no judgments. The connection between averaged judgments and standard deviation is plotted in Figure A.6.

When analyzing the concept pairs with the lowest deviation, there is a clear trend towards particularly highly related pairs, e.g. HYPERNYMY: *Universität – Bildungseinrichtung* (*university – educational institution*); functional relation: *Tätigkeit – ausführen* (*task – perform*); or pairs that are obviously not connected, e.g. *logisch – Juni* (*logical – June*). Table A.3 lists some example concept pairs along with averaged judgments ($\varnothing$) and standard deviation (σ).

Concept pairs with high deviations between judgments often contain polysemous words. For example, *Quelle* (*source*) was disambiguated to *Wasserquelle* (*spring*) and paired with *Text* (*text*). The data shows a clear distinction between one group that rated the pair low (0) and another group that rated the pair high (3 or 4). The latter group obviously missed the point that *textual source* was not an option here. High deviations were also common among special technical terms like (*Mips – Core*), proper names (*Georg – August* – two common first names in German) or functionally related pairs (*agieren – mobil*). Human experience and cultural background clearly influence the judgment of such pairs.

The effect observed here and the effect noted by Budanitsky and Hirst (2006) is

Pair				
German	**English**	**Corpus**	$\varnothing$	σ
Universität – Bildungseinrichtung	university – educational institution	GIRT	3.90	0.30
Tätigkeit – ausführen	task – to perform	BN	3.67	0.58
strafen – Paragraph	to punish – paragraph	GIRT	3.00	1.18
Quelle – Text	spring – text	GIRT	2.43	1.57
Mips – Core	mips – core	SPP	2.10	1.55
elektronisch – neu	electronic – new	GIRT	1.71	1.15
verarbeiten – dichten	to manipulate – to caulk	BN	1.29	1.42
Leopold – Institut	Leopold – institute	SPP	0.81	1.25
Outfit – Strom	outfit – electricity	GIRT	0.24	0.44
logisch – Juni	logical – June	SPP	0.14	0.48

Table A.3: Example concept pairs with averaged judgments and standard deviation. Only one sense is listed for polysemous words. Conceptual glosses are omitted due to space limitations.

probably caused by the same underlying principle. Human agreement on semantic relatedness is only reliable if two words or concepts are highly related or almost unrelated. Intuitively, this means that classifying word pairs as related or unrelated is much easier than numerically rating semantic relatedness. For an information retrieval task, such a classification might be sufficient.

Differences in correlation coefficients for the three corpora are not statistically significant indicating that the phenomenon is not domain-specific. Differences in correlation coefficients for different parts-of-speech are statistically significant (see Table A.2). Verb-verb (VV) and verb-adjective (VA) pairs have the lowest correlation. A high fraction of these pairs is in the problematic medium relatedness area. Adjective-adjective pairs have the highest correlation. Most of these pairs are either highly related or not related at all.

A.3 DKPro

The *Darmstadt Knowledge Processing Software Repository* (DKPro) contains flexible, robust and scalable components for various tasks related to natural language processing, e.g. information retrieval (Müller et al., 2008b) or processing of user generated discourse (Eckart de Castilho and Gurevych, 2009). DKPro builds on the Unstructured Information Management Architecture (UIMA) (Ferrucci and Lally, 2004). Already published components of DKPro are available at `http://www.ukp.tu-darmstadt.de/software/dkpro`. In this thesis, we made use of basic preprocessing components provided by DKPro, but augmented it with two sets of interoperable components for (i) computing semantic relatedness and (ii) keyphrase extraction. The framework for keyphrase extraction was already described in Section 7.1. In the following, we are going to describe the DKPro components developed for computing semantic relatedness.

Figure A.7 visualizes the DKPro pipeline used for the semantic relatedness experiments in this thesis. For both evaluation tasks very similar pipelines are used. They only differ in the input of the evaluation datasets and output of the final eval-

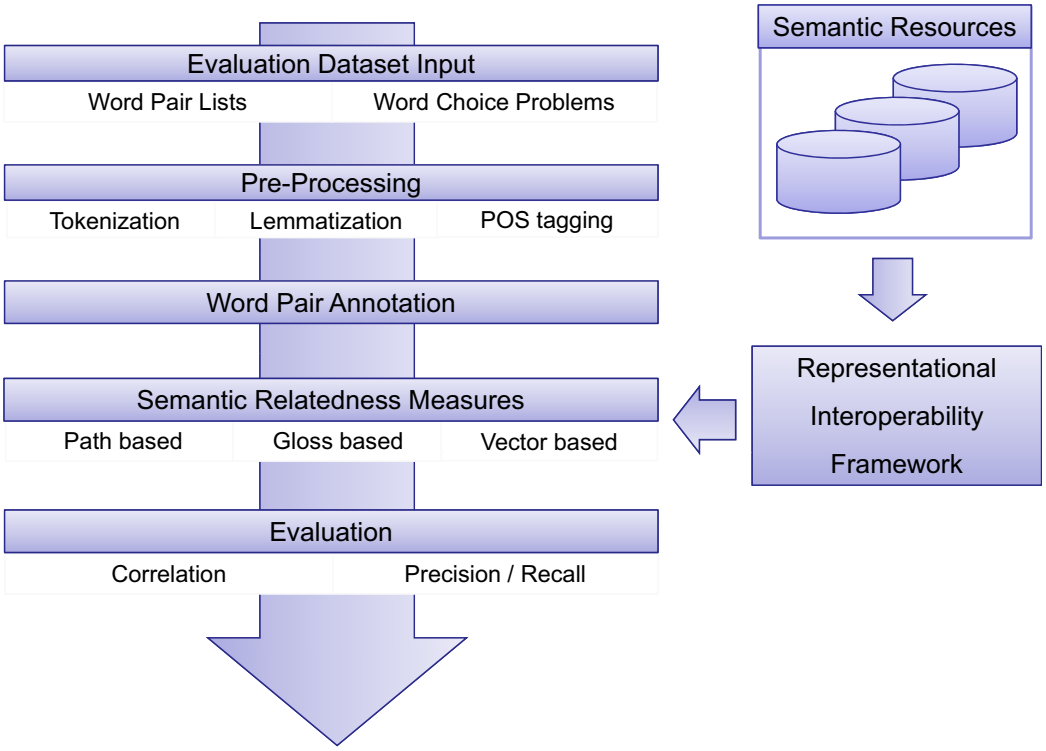


Figure A.7: DKPro pipeline for semantic relatedness experiments.

uation results. For comparison with human judgments, we have to read word pair lists, while for solving word choice problems, we have to read word choice problems. For both cases, there is a specialized reader that expects a standardized format for representing word pair lists and word choice problems. As the two tasks are evaluated differently, we also have two different evaluation modules: one that computes correlation coefficients for the word pair lists and one that computes precision, recall, and F-measure for the word choice problems. The steps in between are the same for both tasks. After some optional pre-precessing steps, we create WordPair annotations between all tokens for which we need to get a semantic relatedness score. In case of word pair lists, we simply create a WordPair annotation for each word pair. For the word choice problems, we create a WordPair annotation between the target word and each candidate token. In the next step, we can have an arbitrary number of semantic relatedness annotators that compute a score for each WordPair annotation. Note that the DKPro components for computing semantic relatedness do not know about the tasks at all. They only know that they read WordPair annotations and that they have to write SemanticRelatedness annotations containing the computed scores for this word pair. The DKPro components for computing semantic relatedness are implemented as wrappers for semantic relatedness measures using the representational interoperability framework. Thus, a certain measure can be used with any semantic resource integrated into the representational interoperability framework. Finally, the evaluation component reads all WordPair and SemanticRelatedness annotations and computes the final evaluation scores.

Appendix B

Result Tables

In this chapter, we show the complete results of our experiments, as we omitted the Pearson correlation scores for the sake of clarity in Chapter 6. These full results may be important for comparison with previous research, e.g. (Budanitsky and Hirst, 2006; Patwardhan and Pedersen, 2006; Strube and Ponzetto, 2006; Gabrilovich and Markovitch, 2007).

(a) Results on English datasets.

Dataset			MC–30		RG–65		Fin1–153		Fin2–200		YP–130	
Word pairs used			30		65		144		190		80	
		Type	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r
	InterAA		-	.90	-	.80	-	.73	-	.55	-	.87
WordNet	Rad89	PL	.84	.78	.84	.80	.41	.32	*.21*	*.21*	.75	.79
	WuP94	PL	.80	.84	.79	.81	.47	.41	*.21*	*.23*	.72	.79
	LC98	PL	.84	.86	.84	.86	.41	.41	*.21*	*.29*	.75	.84
	Res95	IC	.79	.80	.81	.84	.49	.47	*.18*	*.24*	.76	.81
	JC97	IC	.88	.89	.84	.87	.48	.44	*.18*	*.24*	.75	.77
	Lin98	IC	.78	.83	.82	.87	.49	.47	*.19*	*.26*	.77	.79
Wikipedia	Rad89	PL	*.33*	.35	.36	.36	.35	.31	*.20*	*.21*	*.09*	*.07*
	WuP94	PL	*.38*	.35	.37	.36	.38	.33	*.19*	*.17*	*.07*	*.10*
	LC98	PL	*.33*	.28	.36	.35	.35	.28	*.20*	*.23*	*.09*	*.11*
	Res95	IC	.54	.49	.43	.36	.28	*.19*	*.20*	*.19*	*.18*	*.19*
	JC97	IC	*.15*	*.21*	*.13*	*.13*	*.07*	*.12*	*.06*	*.04*	*.05*	*.11*
	Lin98	IC	.55	.52	.45	.44	*.24*	*.00*	*.22*	*.05*	*.20*	*.21*

(b) Results on German datasets.

Dataset			Gur–30		Gur–65		Gur–350	
Word pairs used			27		53		51	
		Type	ρ	r	ρ	r	ρ	r
	InterAA		-	-	-	.81	-	.69
GermaNet	Rad89	PL	.68	.72	.69	.75	.43	.37
	WuP94	PL	.40	.51	.49	.60	*.31*	.38
	LC98	PL	.68	.72	.69	.75	.43	.42
	Res95	IC	.56	.53	.54	.52	.37	*.30*
	JC97	IC	.61	.65	.50	.58	*.23*	*.24*
	Lin98	IC	.56	.55	.54	.54	.37	*.32*
Wikipedia	Rad89	PL	.57	.60	.38	.43	.39	.40
	WuP94	PL	.62	.63	.37	.40	.38	.39
	LC98	PL	.57	.63	.38	.46	.39	.39
	Res95	IC	.63	.60	.41	.46	.42	.38
	JC97	IC	.59	.63	*.32*	.39	*.36*	.37
	Lin98	IC	.63	.62	.40	.46	.41	.38

Table B.1: Correlations of path based (PL) and information content based (IC) measures with human judgments on English and German datasets. Non-significant correlations are in italics (two tailed t-test, $\alpha = .05$).

(a) Results on English datasets.

Dataset		MC–30		RG–65		Fin1–153		Fin2–200		YP–130	
Word pairs used		30		65		146		193		90	
		ρ	r	ρ	r	ρ	r	ρ	r	ρ	r
	InterAA	-	.90	-	.80	-	.73	-	.55	-	.87
WordNet	Les86	.43	.53	.53	.64	*.20*	.30	*.01*	*.07*	.54	.66
WordNet-pseudo	Gur05	.82	.82	.78	.79	.47	.44	.32	.28	.78	.83
Wiktionary	Les86	*.26*	*.15*	*.21*	*.23*	*.21*	.27	*.17*	*.16*	*.10*	*.14*
Wiktionary-pseudo	Gur05	.50	.45	.65	.64	.35	.28	.19	.19	.24	.26
Wikipedia	Les86	.38	.44	*.24*	*.35*	.26	.28	*.09*	*.12*	*.15*	*.14*
Wikipedia-first	Les86	*.17*	*.20*	*.17*	*.24*	*.18*	*.18*	*.12*	*.10*	*.07*	*.08*

(b) Results on German datasets.

Dataset		Gur–30		Gur–65		Gur–350	
Word pairs used		22		39		115	
		ρ	r	ρ	r	ρ	r
	InterAA	-	-	-	.81	-	.69
GermaNet	Les86	.67	.57	.68	.62	.40	.32
Wiktionary	Les86	*.02*	*.11*	*.10*	*.17*	*.01*	*.05*
Wiktionary-pseudo	Gur05	.75	.58	.74	.60	.45	.34
Wikipedia	Les86	*.04*	*.10*	*.19*	*.26*	.35	.34
Wikipedia-first	Les86	*.02*	*.10*	*.01*	*.05*	.27	.27

Table B.2: Correlations of gloss based measures with human judgments on English and German datasets. Non-significant correlations are in italics (two tailed t-test, $\alpha = .05$).

(a) Results on English datasets.

Dataset		MC–30		RG–65		Fin1–153		Fin2–200		YP–130	
Word pairs used		30		65		144		187		90	
		ρ	r	ρ	r	ρ	r	ρ	r	ρ	r
	InterAA	-	.90	-	.80	-	.73	-	.55	-	.87
WordNet	ZG07	.77	.44	.82	.49	.59	.33	.48	.47	.73	.64
Wiktionary	ZG07	.84	.72	.81	.72	.67	.39	.54	.49	.63	.56
Wikipedia	GM07	.72	.66	.75	.66	.67	.48	.38	.37	.29	.31
Wikipedia-first	ZG07	.67	.46	.73	.49	.68	.31	.51	.39	.31	.34
WikipediaLink	M07/NHN07	.45	.49	.56	.56	.60	.57	.45	.44	*.00*	*.01*

(b) Results on German datasets.

Dataset		Gur–30		Gur–65		Gur–350	
Word pairs used		24		46		126	
		ρ	r	ρ	r	ρ	r
	InterAA	-	-	-	.81	-	.69
GermaNet	ZG07	.69	.60	.70	.62	.59	.49
Wiktionary	ZG07	.87	.65	.86	.66	.66	.54
Wikipedia	GM07	.80	.59	.73	.54	.66	.50
Wikipedia-first	ZG07	.53	.44	.57	.36	.61	.31
WikipediaLink	M07/NHN07	.58	.50	.37	.42	.36	.37

Table B.3: Correlations of vector based measures with human judgments on English and German datasets. Non-significant correlations are in italics (two tailed t-test, $\alpha = .05$).