An Evaluation Procedure for Word Net Based Lexical Chaining: Methods and Issues

Irene Cramer and Marc Finthammer

Faculty of Cultural Studies, University of Dortmund, Germany
irene.cramer|marc.finthammer@uni-dortmund.de

Abstract. Lexical chaining is regarded to be a valuable resource for NLP applications, such as automatic text summarization or topic detection. Typically, lexical chainers use a word net to compute semantically motivated partial text representations. However, their output is normally evaluated with respect to an application since generic evaluation criteria have not yet been determined and systematically applied. This paper presents a new evaluation procedure meant to address this issue and provide insight into the chaining process. Furthermore, the paper exemplarily demonstrates its application for a lexical chainer using GermaNet as a resource.

1 Project Context and Motivation

Converting linear text documents into documents publishable in a hypertext environment is a complex task requiring methods for the segmentation, reorganization, and linking. The HyTex project, funded by the DFG, aims at the development of conversion strategies based on text-grammatical features[1]. One focus of our work is on topic-based linking strategies using lexical and thematic chains. In contrast to the lexical ones thematic chains are based on a selection of central words, so called topic anchors, which are e.g. words able to outline the content of a complete passage, and as in lexical chaining connected via semantically meaningful edges. An illustration is given in Fig. 1.

We intend to use lexical chaining for the construction of thematic chains: on the one hand as a feature for the extraction of topic anchors and on the other hand as a tool for the calculation of thematic structure, as shown in Fig. 1. For this purpose, we implemented a lexical chainer for German corpora based on GermaNet. In order to perform an in-depth analysis and evaluation of this chainer as well as to gain insight into the whole chaining process we developed a detailed evaluation procedure. We argue that this procedure is applicable to any lexical chainer regardless of the algorithm or resources used and helps to fine-tune the parameter setting ideal for a specific application. We also present a detailed evaluation of our own lexical chainer and illustrate the issues and challenges we encountered using GermaNet as a resource.

Paper plan: The remainder of this paper is structured as follows: Section 2 describes the basic aspects of lexical chaining and presents a detailed, new evaluation

[1] See our project web pages http://www.hytex.info/ for more information about the concept of thematic chains and the project context.

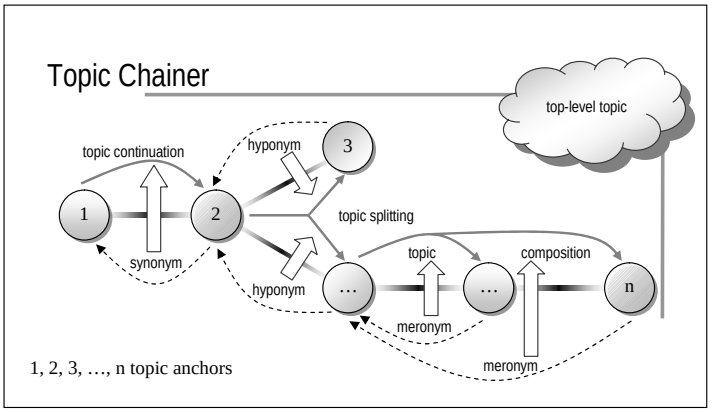

Fig. 1. Topic chaining example

procedure. Section 3 presents the resources used for our lexical chainer and the evaluation. Section 4 discusses our preprocessing component necessary to handle the rather complex German morphology and well-known challenges, such as proper names, in lexical chaining. Section 5 discusses our chaining based disambiguation experiments. Section 6 presents a short overview of eight semantic relatedness measures and compares their values with the results of a human judgment experiment that we conducted. Section 7 outlines the evaluation of our chaining with respect to our application scenario and the project context. Section 8 summarizes and concludes the paper.

2 Lexical Chaining

Based on the concept of lexical cohesion (Halliday and Hasan, 1976) computational linguists e.g. (Morris and Hirst, 1991) developed a method to compute partial text representations: **lexical chains**. To illustrate the idea an annotation is given as an example in Fig. 2. It shows that lexical chaining is achieved by the selection of vocabulary and significantly accounts for the cohesive structure of a text passage. The chains span over passages linking lexical items, where the linking is based on the semantic relations existing between them. Typical semantic relations considered in this context are synonymy, antonymy, hyponymy, hypernymy, meronymy and holonymy as well as complex combinations of these which are computed on the basis of lexical semantic resources such as WordNet (Fellbaum, 1998). In addition to WordNet, which has been used in the majority of cases e.g. (Hirst and St-Onge, 1998), (Green, 1999), (Teich and Fankhauser, 2004), Roget's Thesaurus (Morris and Hirst, 1991) and GermaNet (Mehler, 2005) have already been applied.

Several natural language applications as text summarization e.g. (Barzilay and Elhadad, 1997), (Silber and McCoy, 2002), malapropism recognition (Hirst and St-Onge,

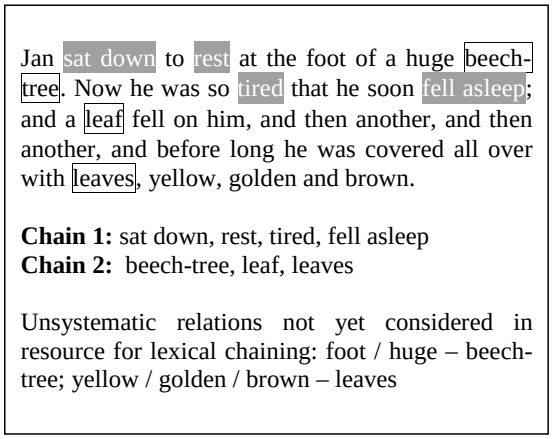

Fig. 2. Chaining example adapted from (Halliday and Hasan, 1976)

1998), automatic hyperlink generation e.g. (Green, 1999), question answering e.g. (Novischi and Moldovan, 2006) and topic detection/topic tracking e.g. (Carthy, 2004) benefit from lexical chains as a valuable text representation.

In this paper we present the evaluation of our own implementation of a lexical chainer for German, `GLexi`, which is based on the algorithms described by (Hirst and St-Onge, 1998) and (Barzilay and Elhadad, 1997) and was developed to support the extraction of thematic structures and topic development. As most systems, `GLexi` consists of the fundamental modules shown in Table 1, which reveals that preprocessing – thus, the selection of the so-called chaining candidates and determination of relevant information about these candidates, like text position and part-of-speech – play a major role in the whole process. A chaining candidate is the fundamental chain element; it is a token comprised of all bits of information belonging to it.

We argue that a sophisticated preprocessing may enhance coverage, which is acknowledged to be a crucial aspect in the development of a lexical chaining system e.g. (Green, 1999), (Barzilay and Elhadad, 1997), and (Hirst and St-Onge, 1998). Accordingly, we address several ideas to improve the coverage of our system. At least two issues independent of language influence this aspect:

- limitations imposed on the whole process by the size and coverage of the lexical semantic resource used,
- and the presence of proper names in the text, which cannot be resolved without extensive preprocessing.

However, it is even more critical for German coverage because

- of its complex morphology (e.g. inflection and word formation)
- and the smaller coverage of GermaNet in comparison to WordNet.

Table 1. Overview of chainer modules

Module	Subtasks
preprocessing of corpora	chaining candidate selection: determine chaining window, sentence boundaries, tokens, POS-tagging, chunks etc.
core chaining algorithm calculation of chains or meta-chains	lexical semantic look-up resource (e.g. WordNet), scoring of relations, sense disambiguation
output creation	rating/scoring of chain strength build application specific representation

Both aspects as well as coverage in general are discussed in detail in the following sections.

In order to formally evaluate the performance – in terms of precision and recall – of `GLexi` for various parameter settings a (preferably standardized and freely available) test set would be required. To our knowledge there is no such resource – neither for English nor for German. Therefore, we have started to investigate the development of such a gold standard for German corpora. Initial results are discussed in (Stührenberg et al., 2007). Our experiments show that the manual annotation of lexical chains is a demanding task, which has also been emphasized in the work by (Morris and Hirst, 2004), (Morris and Hirst, 2005) and (Beigman Klebanov, 2005). The rich interaction between various principles to achieve a cohesive text structure seems to distract annotators. We therefore argue that the evaluation of a lexical chainer might be best performed in four steps:

- **evaluation of coverage**: amount of chaining candidates the chainer is able to process,
- **evaluation of disambiguation quality**: number of chaining candidates correctly disambiguated with respect to lexical semantic resource,
- **evaluation of quality of semantic relatedness measures**: comparison with human judgment,
- **evaluation of chains** with respect to concrete application.

This procedure ensures that the most relevant parameters in the evaluation of our system, `GLexi`, can be judged separately and also enables us to gain the necessary insight into the chaining process.

3 Resources

We based the evaluation of our system and all experiments described in this paper on three main resources: GermaNet as the lexical semantic lexicon for our chainer, the

HyTex project corpus and a set of word pairs compiled in a human judgment experiment for the evaluation steps discussed in Sect. 6.2.

3.1 GermaNet

GermaNet (Lemnitzer and Kunze, 2002b) is a machine readable lexical semantic lexicon for the German language developed in 1996 within the LSD Project at the Division of Computational Linguistics of the Linguistics Department at the University of Tübingen. Version 5.0 covers approximately 77,000 lexical units – nouns, verbs, adjectives and adverbs as well as some multi word units – grouped into approximately 53,500 so-called synonym sets. GermaNet contains approximately 4,000 lexical (between lexical units) and approximately 64,000 conceptual (between synonym sets) connections. Although it has much in common with the English WordNet (Fellbaum, 1998) there are some differences; see (Lemnitzer and Kunze, 2002a) for more information about this issue. The most important difference in our opinion is the fact that GermaNet is much smaller than WordNet, which has a negative impact on the coverage. However, we found that none of the other differences, such as the presence of artificial concepts, have much influence over the results of our chainer.

3.2 Corpus

For the evaluation steps mentioned in Sect. 2 we used a part of the HyTex corpus, which contains 130 documents (approximately 3 million words). It was compiled and in parts manually annotated in project phase I; see (Beißwenger and Wellinghoff, 2006) for more information. The HyTex corpus consistis of 3 subcorpora: the so-called *core corpus*, *supplementary corpus* and *statistics corpus*. The corpora contain scientific papers, technical specifications, tutorials and textbook chapters, as well as FAQs about language technology and hypertext research.

In the core corpus logical text structure is marked, for example the organization of documents into chapters, sections, passages, figures, footnotes, tables etc. is annotated using DocBook-based XML tags; see (Lenz and Lüngen, 2004) for more information. In order to split the documents into *chainable* sections, we used the core corpus and segmented the documents according to its annotation. The homogeneity and relevance of a chain largely depends on its length and thus on the length of the underlying text. We found the average length of a section to be adequate for chaining of our domain-specific corpus. We also decided to only select nouns and noun phrases as chaining candidates because our experiments revealed that terminology plays the key role in scientific and technical documents terminology.

3.3 Set of Word Pairs

In order to evaluate the quality of a relatedness measure, a set of pre-classified word pairs (in our case for German) is necessary. In previous work for English, most researchers used Rubenstein and Goodenough's list (Rubenstein and Goodenough, 1965) or Miller and Charles's list (Miller and Charles, 1991). For German there are – to our

knowledge – three sets of word pairs: a translation of Rubenstein and Goodenough's list by (Gurevych, 2005), a manually generated set of 350 word pairs by (Gurevych and Niederlich, 2005), and a semi-automatically generated set by (Zesch and Gurevych, 2006). Unfortunately, we could not find any of these German sets published. We also argue that the translation of a list constructed originally for English subjects might bias the results and therefore decided to compile our own set of word pairs as can be seen in Table 2. The goal was to cover a wide range of relatedness types, i.e. systematic and unsystematic relations, and relatedness levels, i.e. various degrees of relation strength. We also included nouns of diverse semantic classes, e.g. abstract nouns, such as `das Wissen` (Engl. knowledge), and concrete nouns, such as `das Bügeleisen` (Engl. flat-iron). We thus constructed a list of approximately 320 word pairs, picked 100 of these to evenly meet the constraints mentioned above and randomized them. We also included words which occur more than once (up to 8 times) in a word pair; these are grouped into consecutive blocks. We asked 35 subjects to rate the word pairs on a 5-level scale (0 = not related to 4 = strongly related). The subjects were instructed to base the rating on their intuition about any kind of conceivable relation between the two words. We used this list and the human judgment to evaluate the semantic relatedness measures described in Sect. 6.1.

4 Evaluation Phase I – Preprocessing Methods

We conducted several experiments to investigate the coverage of GermaNet and thus the coverage of `GLexi`. We found that GermaNet contains 56.42% of the 28,772 noun tokens mentioned in the corpus. We concluded from a sample analyzed that this coverage issue stems from the rich German morphology, domain-specific terminology and proper names, which are both not covered sufficiently by GermaNet. We therefore implemented the preprocessing architecture shown in Fig. 3. A document is first segmented into sections and then split into sentences and tokens. In addition, for each token a list of features is extracted, such as position in the document (with respect to sentence and section), part-of-speech, lemma, and morphology[2]. On this basis the preprocessing component generates one or several alternative chaining candidates, e.g. the first alternative would be the singular instead of a plural, like for cats ⇒ cat. The second alternative considers compounds when applicable. Since our corpus is very rich in compounds this plays a major role in the implementation of our system and is discussed in more detail in Sect. 4.1 Technical terminology and proper names are also considered separately as alternatives.

4.1 German Morphology

Compared to English, the German noun morphology is relatively complex: especially the presence of four cases and compounds, which are written as one word and not divided by blanks, plays a major role in our chaining system.

[2] For our study we used the Insight Discoverer^{TM}Extractor Version 2.1. (cf. http://www.temis-group.com/). We thank the TEMIS group for kindly permitting us to use this technology in the framework of our project.

Table 2. Word pairs and human judgment mean value

Word 1	Word 2	Mean Value
Nahrungsmittel	Essen	3.94
Wasser	Flüssigkeit	3.94
Eltern	Kind	3.86
Blume	Pflanze	3.86
Angst	Furcht	3.86
Kamin	Schornstein	3.80
Blume	Tulpe	3.80
Sonne	Sommer	3.71
Blume	Duft	3.69
Wasser	Fisch	3.69
Mensch	Lebewesen	3.66
Schwimmbad	Bademeister	3.63
Riese	Gigant	3.63
Mitarbeiter	Kollege	3.60
Behandlung	Therapie	3.54
Lampe	Leuchte	3.49
Entdecker	Expedition	3.49
Ozean	Tiefe	3.46
Wahl	Demokratie	3.43
Badekappe	Schwimmer	3.40
Würfel	Zufall	3.37
Wissen	Kenntnis	3.34
Schwimmbad	Becken	3.31
Würfel	Spiel	3.31
Nahrungsmittel	Hunger	3.31
Bewegung	Tanz	3.26
Kälte	Wärme	3.20
Mensch	Verstand	3.20
Nahrungsmittel	Restaurant	3.20
Wissen	Schule	3.17
Zuverlässigkeit	Freundschaft	3.17
Politiker	Bürgermeister	3.17
Wissen	Quiz	3.09
Blume	Wasser	3.09
Herbst	Winter	3.03
Kontinent	Landkarte	3.03
Sonne	Leben	3.00
Wissen	Intelligenz	3.00
Märchen	Geschichte	2.94
Sonne	Stern	2.91
Unterhaltung	Programm	2.91
Etage	Wohnung	2.83
Wasser	Pirat	2.80
Treppe	Aufzug	2.77
Haushalt	Ordnung	2.74
Blume	Honig	2.74
Blume	Liebe	2.71
Nahrungsmittel	Händler	2.66
Mensch	Krankheit	2.57
Tür	Fenster	2.54

Word 1	Word 2	Mean Value
Sonne	Strom	2.51
Wasser	Nebel	2.49
Wasser	Trockenheit	2.43
Schwimmbad	Ferien	2.40
Kino	Theater	2.40
Nahrungsmittel	Tier	2.34
Wissen	Alter	2.31
Würfel	Mathematik	2.23
Mensch	Hund	1.91
Wasser	Palme	1.89
Schwimmbad	Ausdauer	1.77
Würfel	Betrug	1.57
Würfel	Kugel	1.49
Nahrungsmittel	Jahreszeit	1.46
Schwimmbad	Eis	1.43
Wüste	Quelle	1.34
Mensch	Weltraum	1.26
Wetter	Hoffnung	1.26
Licht	Bremse	1.17
Nahrungsmittel	Zahn	1.11
Schwimmbad	Stadt	1.09
Wissen	Vergnügen	1.03
Beschleunigung	Lautstärke	1.03
Geographie	System	0.80
Computer	Hotel	0.71
Pflanze	Klebstoff	0.54
Datum	Auslastung	0.54
Sonne	Arzt	0.31
Glaube	Rennen	0.29
Mensch	Wolke	0.20
Sonne	Dirigent	0.17
Nation	Garten	0.17
Mittagessen	Becken	0.17
Farbe	Richter	0.14
Volk	Punkt	0.11
Richtung	Lied	0.11
Schleuder	Schallplatte	0.09
Löffel	Baum	0.09
Nahrungsmittel	Kabel	0.09
Hitze	Familie	0.09
Wasser	Rundfunk	0.09
Rausch	Monat	0.06
Tasse	Motor	0.03
Dach	Wal	0.03
Schwimmbad	Gabel	0.03
Gardine	Bleistift	0.03
Oase	Bügeleisen	0.03
Wäscheleine	Toastbrot	0.03
Würfel	Wasser	0.03
Flosse	Drucker	0.00

Table 3. Coverage of GermaNet

The approximately 29,000 (noun) tokens in our corpus split into			
56% in GermaNet	44% not in GermaNet, of these:		
	15% inflected	12% compounds	17% small, uncovered classes (see Table 3)

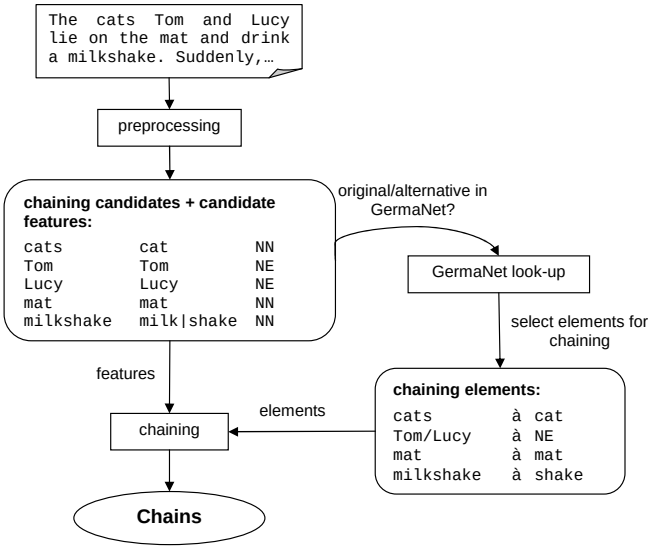

Fig. 3. Preprocessing architecture

Notes on German inflection: In order to ensure that inflected nouns can be handled accurately we rely on lemmatization. Inflection in German means four cases and singular/plural forms.

Coverage improvement on the basis of inflection processing: On the basis of our lemmatization step, we were able to replace approximately 15% of the nouns by their lemmata and could thus increase the coverage to 71%.

Open Issues: However, we found that there are some cases in which the original (plural) form in the text should not be normalized to its singular form, e.g. the German word `Daten` (Engl. data or dates) can be lemmatized to `Datum` (Engl. date); the same holds for `Medien` (Engl. media) and `Medium` (Engl. psychic, data carrier). Thus, when lemmatized the words change their meaning. Moreover, the plural form is not included in GermaNet. Consequently, our system uses as a chaining element the first alternative of the original, e.g. `Datum` instead of `Daten`. Of course, in our domain specific corpus `Daten` (Engl. data) and `Medien` (Engl. media) are frequent words (`Daten` occurred 78 times in the corpus, `Medien` 41 times), which serve in the chains as glue for a list of other chaining elements and therefore need to be carefully considered. In addition, lemmatization is not very reliable for compounds. Nevertheless, we think that the results mentioned above emphasize that this preprocessing step is a necessary aspect to improve the coverage of a baseline chaining system.

Notes on German compounds: Compounds are frequent in our limited domain corpus. Two or more (free) morphemes are combined into one word, the compound,

e.g. Druckerpatrone (components: Drucker and Patrone; Engl. ink catridge). Sometimes, the components are additionally joined by a so-called *Fugenelement* (Engl. gap element), e.g. Liebeslied (components: Liebe and Lied, gap element: s; Engl. love song). Typically, the complete compound inherits the grammatical features, such as genus, of its last – so-called head – component, thus the one at the rightmost position, e.g. das Lied (genus: neutral; Engl. song) and das Liebeslied (genus: neutral), while it is die Liebe (genus: feminine; Engl. love). In addition to these grammatical features of compounds in German there are at least two semantically motivated classes: the semantically transparent and the intransparent compounds. Semantically transparent describes a compound for which the meaning of the whole can be deduced from the meaning of its parts, e.g. a Liebeslied (Engl. love song) is a kind of Lied (this component is the head of the compound; Engl. song), where the component Liebe (Engl. love) can be seen as the modifier of the head. In contrast, the meaning of a semantically intransparent compound cannot be deduced from its parts, e.g. Rotkehlchen (Engl. robbin; components: rot, Engl. red, and Kehlchen, which can be split into Kehle, Engl. throat and -chen diminutive suffix). An ideal lexical semantic resource would cover all intransparent compounds, whereas the transparent ones would not necessarily be included since it is possible to derive their meaning intellectually or automatically. In principle GermaNet accounts for this rule, however, there are as always some compounds which are not included.

Coverage improvement on the basis of compound processing: On the basis of the morphological analysis we were able to include previously uncovered words, i.e. approximately 12% of the nouns could be replaced by their compound head word (e.g. Liebeslied would be replaced with Lied) and thus increase the coverage to 83%.

Open Issues: However, this step has at least two major drawbacks. First, the morphological analysis generated by the Insight Discoverer^{TM}Extractor Version 2.1 contains all possible readings, e.g. the German word Agrarproduktion (Engl. agricultural production) might be split among other things into Agrar (Engl. agricultural), Produkt (Engl. artifact) and Ion (Engl. ion [chem.]). The automatic selection of a correct reading is in some cases demanding and the effect on the whole chaining process might be severe – e.g. given the word Produktion and the morphological analysis mentioned the chainer could decide to replace the word Produktion, given it cannot be found in GermaNet, with the word Ion, which could completely mislead the disambiguation of word sense in the chaining and thus the whole chaining process itself. Second, compounds containing more than two components could be split into several headwords, e.g. the head-word of the compound Datenbankbenutzerschnittstelle (Engl. data base user interface) could be Benutzerschnittstelle (Engl. user interface) or Schnittstelle (Engl. interface) or even only Stelle (Engl. position or area[3]). In our future work, we therefore plan to investigate which parameter settings might be ideal on the one hand to improve the coverage and on the other hand to account for semantic disambiguation performance. Nevertheless, we think that morphological analysis of compounds is a crucial aspect in the preprocessing of our lexical chainer.

[3] Note: This is the correct though in this context semantically inadequate translation.

4.2 Smaller Classes of Uncovered Material

As Table 3 shows, with our first preprocessing step we were able to include approximately 27% of the words, which we could initially not find in GermaNet, i.e. approximately 15% on the basis of lemmatization and approximately 12% on the basis of compound analysis. We examined a sample of the remaining 17%, the results are shown in Table 4. We found in the sample approximately 15% proper names, approximately 30% foreign words, especially technical terminology in English, approximately 25% abbreviations, and approximately 20% nominalized verbs, which are not sufficiently included in GermaNet and very prominent in German technical documents. The rest (not shown in Table 4) consists of incorrectly tokenized or POS-tagged material, such as broken web links.

Table 4. Detailed analysis of small classes not covered by GermaNet

The small, uncovered classes (see Table 2) split into			
15% proper names	30% foreign words	25% abbreviations	20% nominalized verbs

No matter which language is considered, proper names are a well-known challenge in lexical chaining, e.g. (Green, 1999). They are semantically central items in most corpora and therefore need to be handled with care. The same holds for technical terminology, in many cases multi-word units, which are obviously very frequent and relevant in technical and academic documents. We deal with both in the second phase of our preprocessing component. However, note that we only treat the classical named entities, i.e. names belonging to people, locations, and organizations. We do not yet cover other proper names.

We included the recognition of proper names and multi-word units in our preprocessing. After the basic preprocessing, such as sentence boundary detection, tokenization and lemmatization, which is accomplished by the Insight Discoverer^{TM}Extractor Version 2.1, we run the second preprocessing phase, which splits into the following two subtasks:

- Proper name recognition and classification: We use a simple named entity recognizer (NER) for German[4], which tags person names, locations, and organizations.
- Simple chunking of multi word units and simple phrases: We use the part-of-speech tags computed in the first preprocessing step by the Insight Discoverer^{TM}Extractor Version 2.1 to construct simple phrases.

Of course, these are interim solutions, and we plan to investigate strategies to improve the second preprocessing phase in our future work. Because we found names of conferences and product names to be relatively frequent, we intend to extend our NER system accordingly. Most of the technical terminology in our corpus is not included

[4] It is our own machine learning based implementation of a simple NER system.

in GermaNet and could thus not be considered in the chaining. However, in the HyTex project we developed a terminological lexicon for our corpus (called TermNet), see (Beißwenger et al., 2003) and (Kunze et al., in this volume), which we plan to use in addition to GermaNet. Ultimately, we hope this will again improve the coverage of our chainer. While it is thus far unclear how to handle nominalized verbs and abbreviations, the statistics shown in Table 4 emphasize their relevance, and they certainly need to be considered with care in our future work.

To conclude, without any preprocessing only 56% of the noun tokens in our corpus are chainable. Approximately 67% of the remaining nouns can be handled with morphological analysis and a very simple NER system. The remaining approximately 33% is comprised of abbreviations, foreign words, nominalized verbs and broken material as well as not yet covered proper names and technical terminology, which we intend to deal with in an expansion of our lexical semantic resource, i.e. in a combination of GermaNet and TermNet, statistical relatedness measures based on web counts and a refinement of our preprocessing components.

5 Evaluation Phase II – Chaining-based Word Sense Disambiguation

In addition to the coverage issues described in Sect. 4 word sense disambiguation has a high impact on the performance of a lexical chainer. That is, if incorrectly disambiguated, a word with several word senses, such as *bank* or *mouse*, could mislead the complete chaining algorithm and cause the construction of inappropriate chains. As a matter of course, the disambiguation performance of a chainer is not able to outperform high-quality WSD systems, such as presented at the Senseval workshops, and it is not our purpose to compete against these systems but to locate potential sources of error in the chaining procedure. Consequently, the second step in our evaluation procedure is related to word sense disambiguation, in our case the selection of an appropriate synonym set in GermaNet. In principle, there are at least two different methods: the greedy selection of a word sense and the subsequent selection. Greedy word sense disambiguation means to choose the first matching synonym set which exhibits a suitable path or a semantic relatedness measure value. In contrast, subsequent disambiguation, see e.g. (Silber and McCoy, 2002), means to first assemble all possible readings, i.e. all in principle suitable paths or semantic relatedness measure values, and then, given this information, select the best match. However, both methods have their pros and cons: the greedy selection is simple and straightforward, but it tends to pick the wrong word sense in cases in which the correct reading of a word cannot be determined until the rest of the potential chaining partners are examined. The subsequent word sense disambiguation supports exactly this issue, but it is rather complex, especially when several relatedness measures are to be considered. In addition to these two methods, there are several ranges between the greedy and the subsequent disambiguation: e.g. the appropriate synonym set of a word might be determined on the basis of a majority vote when all possible combinations containing this word are read. Alternatively, the information content (see Sect. 6.1) might be useful to pick a word sense.

Analysis of the chaining-based word sense disambiguation: In lexical chaining, the disambiguation is essentially based on the selection of a word sense with respect to a path or relatedness measure value between synonym sets. For example, a pair of words A, with three senses, and B, with two senses, has six possible readings: thus, the probability to pick the correct one is only $1/6$. The more senses a word pair exhibits, the likelier it is to pick an incorrect reading for at least one of the two words. Table 5 shows the distribution of word senses for the (noun) tokens[5] in our corpus. Obviously, almost every second token features more than one word sense in GermaNet. That means in the worst case every second token can in principle mislead the chainer in the case of an incorrect disambiguation.

Table 5. Overview of the number of word senses occurring in our corpus

1 sense	2 senses	3 senses	4 senses	$>$ 4 senses
$\sim$ 53%	$\sim$ 22%	$\sim$ 15%	$\sim$ 7%	$\sim$ 3%

word A	word B	word sense	word sense	Wu-Palmer value	rank
Text	Hypertext	1	1	0,9231	1
Text	Hypertext	2	1	0,8333	2
manually annotated word sense (correct word sense)					
Text	Hypertext	1	1		
à best Wu-Palmer value – correct word sense (rank 1)					

Fig. 4. Example ranking of the various readings

However, it is the basic idea of lexical chaining that lexicalized coherence in the text accounts for the mutually correct disambiguation of the words in a pair. In order to investigate the disambiguation quality, we randomly selected a corpus sample and computed the relatedness values. We then ranked the possible readings for each word pair according to their relatedness values. An example is shown in Fig. 4. We evaluated this against our manual annotation of word senses. The results are shown in Table 6. The three best relatedness measures in this context, Resnik, Wu-Palmer and Lin, correctly

[5] We consider tokens instead of types because in principle every single occurrence of a word might exhibit a different word sense. We have such examples in our corpus, e.g. in one sentence the word `text` is used with three different senses.

disambiguate approximately 50% of the word pairs in our sample. For all eight measures the correct reading is on the first four ranks in the majority of the cases. Although this disambiguation accuracy is only mediocre, it outperforms the baseline (approximately 39% correct disambiguation on rank 1), i.e. the performance of a chainer using the information content of a word to disambiguate its word sense. As mentioned above an additional alternative method to select the correct word sense is the majority voting: for a list of word pairs with one given word and all possible chaining partners in the text (e.g. mouse - computer, mouse - hardware, mouse - keyboard, mouse - etc.), the word sense, which is supported by most of the top-ranked relatedness measure values, is supposed to be the correct one. Our experiments showed that a majority voting is able to enhance the accuracy and bring the rate in some cases up to 63% correct disambiguation. We plan to investigate in our future work how we can again improve the disambiguation quality of our chainer. We especially plan to explore the method of meta-chaining proposed in (Silber and McCoy, 2002) and to adapt it for a multiple relatedness measure chaining framework. In addition, the integration of a WSD system might positively influence the performance of our chainer.

Table 6. Overview of semantic relatedness-based disambiguation performance

correct disamb. on	Graph Path	Tree Path	Wu-Palmer	Leacock-Chodorow
rank 1	34.93%	42.13%	50.67%	34.93%
rank 1 – 4	79.20%	80.80%	86.40%	79.20%
	Hirst-StOnge	Resnik	Jiang-Conrath	Lin
rank 1	17.07%	57.60%	37.60%	50.13%
rank 1 – 4	19.20%	88.80%	77.87%	87.20%

6 Evaluation Phase III – Semantic Relatedness and Similarity

The third step in our evaluation procedure is related to the semantic measures, which are calculated on the basis of a lexical semantic resource (and word frequency counts) and used in the construction of lexical chains. A semantic measure expresses *how much two words have to do with each other*. The notion of semantic measure is controversially discussed in the literature e.g. (Budanitsky and Hirst, 2001). The two most relevant terms in this context are semantic similarity and semantic relatedness, defined according to (Budanitsky and Hirst, 2001) as follows:

- Semantic similarity: Word pairs are considered to be semantically similar if any synonymy or hypernymy relations hold. (Examples: forest - wood ⇒ synonymy, flower - rose ⇒ hypernymy, rose - oak ⇒ common hypernym: plant)
- Semantic relatedness: Word pairs are considered to be semantically related if any systematic relation, such as synonymy, antonymy, hypernymy, holonymy, or any unsystematic relation holds. Compared to the semantic similarity measures this is

the more general concept, as it includes any intuitive association or linguistically formalized relation between words. (Examples: flower - gardener or monkey - banana ⇒ intuitive association, tree - branch ⇒ holonymy, day - night ⇒ antonymy)

According to the definition by (Budanitsky and Hirst, 2001), semantic similarity is a subtype of semantic relatedness; in the following section we discuss various relatedness measures. In order to explore these measures and their relevant characteristics, we used the results of our human judgment experiment described in Sect. 3.3.

6.1 GermaNet-based Semantic Relatedness Measures

We expect that good lexical chains include systematic and unsystematic relations, a position which has also been stressed by the experiments reported in (Morris and Hirst, 2004) and (Morris and Hirst, 2005). In fact, most of the established measures merely consider synonymy and hypernymy. Therefore, they actually fall under the notion of semantic similarity.

Figure 5 outlines how the calculation of the relatedness measures interacts with the chaining algorithm and the semantic resource. When the preprocessing is completed, the chaining algorithm selects chaining candidate pairs, in other words, word pairs, for which the relatedness needs to be determined (see Fig. 5 – *Query 1: relatedness of word A and B?*). Next, the relatedness measure component (RM component) performs a look-up in the semantic resource in order to extract all available features, such as shortest path length or information content of a word, which are necessary to calculate the relatedness value (see Fig. 5 – *Query 2: semantic information about A and B?*). On the basis of these features, the RM component computes a value which represents the strength of the semantic relation between the two words.

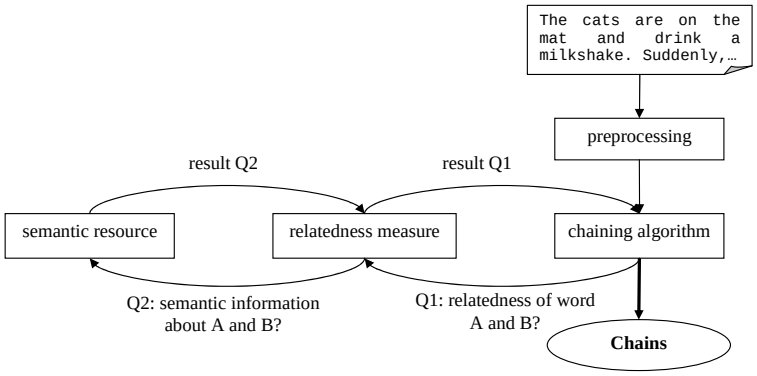

Fig. 5. Use of relatedness measures in chaining

The various measures introduced in the literature use different features and therefore also cover different concepts or aspects of semantic relatedness. We have implemented eight of these measures, which are shortly sketched out below. All eight measures are based on a lexical semantic resource, in our case GermaNet, and some additionally utilize a word frequency list[6].

The first four measures use a hyponym-tree induced from GermaNet. That means, given GermaNet represented as a graph, we exclude all edges except the hyponyms. Since this gives us a wood of nine trees, we then connect them to an artificial root and thus construct the required GermaNet hyponym-tree.

– **Leacock-Chodorow** (Leacock and Chodorow, 1998): Given a hyponym-tree, the Leacock-Chodorow measure computes the length of the shortest path between two synonym sets and scales it by the depth of the complete tree.

$$\mathrm{rel}_{\mathrm{LC}}(s_1, s_2) = -\log \frac{2 \cdot \mathrm{sp}(s_1, s_2)}{2 \cdot D_{Tree}} \tag{1}$$

s_1 and s_2: the two synonym sets examined; $\mathrm{sp}(s_1, s_2)$: length of shortest path between s_1 and s_2 in hyponym-tree; D_{Tree}: depth of the hyponym-tree

– **Wu-Palmer** (Wu and Palmer, 1994): Given a hyponym-tree, the Wu-Palmer measure utilizes the least common subsumer in order to compute the similarity between two synonym sets. The least common subsumer is the deepest vertex which is a direct or indirect hypernym of both synonym sets.

$$\mathrm{rel}_{\mathrm{WP}}(s_1, s_2) = \frac{2 \cdot \mathrm{depth}(\mathrm{lcs}(s_1, s_2))}{\mathrm{depth}(s_1) + \mathrm{depth}(s_2)} \tag{2}$$

$\mathrm{depth}(s)$: length of the shortest path form root to vertex s; $\mathrm{lcs}(s)$: least common subsumer of s

– **Resnik** (Resnik, 1995): Given a hyponym-tree and frequency list, the Resnik measure utilizes the information content in order to compute the similarity between two synonym sets. As typically defined in Information Theory, the information content is the negative logarithm of the probability. Here the probability is calculated on the basis of subsumed frequencies. A subsumed frequency of a synonym set is the sum of frequencies of the set of *all* words which are in this synonym set, *or* a direct or indirect hyponym synonym set.

$$\mathrm{p}(s) := \frac{\sum_{w \in W(s)} \mathrm{freq}(w)}{\mathit{TotalFreq}} \tag{3}$$

$$\mathrm{IC}(s) := -\log \mathrm{p}(s) \tag{4}$$

$$\mathrm{rel}_{\mathrm{Res}}(s_1, s_2) = \mathrm{IC}(\mathrm{lcs}(s_1, s_2)) \tag{5}$$

$\mathrm{freq}(w)$: frequency of a word within a corpus; $W(s)$: set of the synonym set s and all its direct/indirect hyponym synonym sets; $\mathit{TotalFreq}$: sum of the frequencies of all words in GermaNet; $\mathrm{IC}(s)$: information content of the synonym set s

[6] We used a word frequency list computed by Dr. Sabine Schulte im Walde on the basis of the Huge German Corpus (see http://www.schulteimwalde.de/resource.html). We thank Dr. Schulte im Walde for kindly permitting us to use this resource in the framework of our project.

– **Jiang-Conrath** (Jiang and Conrath, 1997): Given a hyponym-tree and frequency list, the Jiang-Conrath measure computes the distance (as opposed to similarity) of two synonym sets. The information content of each synonym set is included separately in this distances value, while the information content of the least common subsumer of the two synonym sets is subtracted.

$$\mathrm{dist}_{\mathrm{JC}}(s_1, s_2) = \mathrm{IC}(s_1) + \mathrm{IC}(s_2) - 2 \cdot \mathrm{IC}(\mathrm{lcs}(s_1, s_2)) \tag{6}$$

– **Lin** (Lin, 1998): Given a hyponym-tree and a frequency list, the Lin measure computes the semantic relatedness of two synonym sets. As the formula clearly shows, the same expressions are used as in Jiang-Conrath. However, the structure is different, as the expressions are divided not subtracted.

$$\mathrm{rel}_{\mathrm{Lin}}(s_1, s_2) = \frac{2 \cdot \mathrm{IC}(\mathrm{lcs}(s_1, s_2))}{\mathrm{IC}(s_1) + \mathrm{IC}(s_2)} \tag{7}$$

– **Hirst-StOnge** (Hirst and St-Onge, 1998): In contrast to the four above-mentioned methods, the Hirst-StOnge measure computes the semantic relatedness on the basis of the whole GermaNet graph structure. It classifies the relations considered into 4 classes: *extra strongly related*, *strongly related*, *medium strongly related*, and *not related*. Two words are considered to be
  - *extra strongly related* if they are identical;
  - *strongly related* if they are synonym, antonym or if one of the two words is part of the other one and additionally a direct relation holds between them;
  - *medium strongly related* if there is a path in GermaNet between the two which is shorter than six edges and matches the patterns defined by (Morris and Hirst, 1991).

  In any other case the two words are considered to be unrelated. The relatedness values in the case of extra strong and strong relations are fixed values, whereas the medium strong relation is calculated based on the path length and the number of changes in direction.
– **Tree-Path** (Baseline 1): Given a hyponym-tree, the simple Tree-Path measure computes the length of a shortest path between two synonym sets. Due to its simplicity, the Tree-Path measure serves as a baseline for more sophisticated similarity measures.

$$\mathrm{dist}_{\mathrm{Tree}}(s_1, s_2) = \mathrm{sp}(s_1, s_2) \tag{8}$$

– **Graph-Path** (Baseline 2): Given the whole GermaNet graph structure, the simple Graph-Path measure calculates the length of a shortest path between two synonym sets in the whole graph, i.e. the path can make use of all relations available in GermaNet. Analogous to the Tree-Path measure, the Graph-Path measure gives us a very rough baseline for other relatedness measures.

$$\mathrm{dist}_{\mathrm{Graph}}(s_1, s_2) = \mathrm{sp}_{Graph}(s_1, s_2) \tag{9}$$

$\mathrm{sp}_{Graph}(s_1, s_2)$: Length of a shortest path between s_1 and s_2 in the GermaNet graph

Differences and Challenges: Most of the measures described in this section are completely based on the hyponym-tree. Therefore, many potentially useful edges of the word net graph structure are not considered, which affects the holonymy (in GermaNet approximately 3,800 edges), meronymy (in GermaNet approximately 900 edges) and antonymy[7] (in GermaNet approximately 1,300 edges) relations. Some of the measures additionally use the least common subsumer. Word pairs featuring potentially different levels of relation are thus subsumed[8]. One could also question if this is the only relevant information to be found in the hyponym-tree for a word pair. Interesting features such as network density or node depth are not included. Moreover, several measures rely on the concept of information content, for which a frequency list is required. Thus, the performance of experiments utilizing different lists as a basis is not directly comparable. Especially for lexical chaining, unsystematic relations are considered to be relevant, see e.g. (Miller and Charles, 1991) and (Morris and Hirst, 2005). However, these are not in GermaNet and consequently cannot be considered in any of the measures mentioned above. We therefore expect them to produce many false negatives, i.e. low relation values for word pairs which are judged by humans to be (strongly) related.

Interpretation of relatedness measure values: Most of the relatedness measures mentioned in Sect. 6.1 are continuous, with the exception of Hirst-StOnge, Tree-Path and Graph-Path which are all discrete. All of the measures range in a specific interval between 0 (not related) and a maximum value, mostly 1. In any case, for each measure the interval could be normalized into a value ranging between 0 and 1. For the three distance measures, Jiang-Conrath, Tree-Path and Graph-Path, a concrete distance value can be converted into its corresponding relatedness value by subtracting it from the theoretical maximum distance. Suppose we plotted the empirically determined relatedness values[9] against ideal relatedness measure values, we would get exemplary distribution functions as shown in Fig. 6a. For a specific empirically determined value, e.g. 0.5, we then obtained different values for the various measures considered, e.g. 0.27 for measure A and 0.94 for measure B. Thus, the values of a specific relatedness measure A range between 1 and approximately 0.94 for an empirically determined interval of relation strengths (e.g. the word pair is strongly related) whereas a relatedness measure B exhibits values between 1 and 0.27 for the same relations. In order to profitably use this information in our chaining system, we need to interpret the values and thus find intervals mapping between e.g. classes of relation strength and measure values[10]. In any case, the distribution functions should be noisy, as shown in Fig. 6b – at best indicating a trend function. However, as Figures 7a–c, 8a–c and 9a–b illustrate, the real values of our eight measures plotted against the empirically determined relatedness values do not display any kind of obvious trend function.

[7] Because antonyms are mostly organized as co-hyponyms, they are – in fact – not completely discarded in the hyponym-tree-based approaches.

[8] Given a pair of words w_A and w_B and their least common subsumer LCS_{AB}, all pairs of a descendant of w_A and a descendant of w_B have LCS_{AB} as their least common subsumer.

[9] These are the values deduced form our human judgment experiment mentioned in Sect. 3.3.

[10] Note that we need to discriminate between the distribution functions (considering empirically determined values and measure values, as exemplarily shown in Fig. 6) and the relatedness

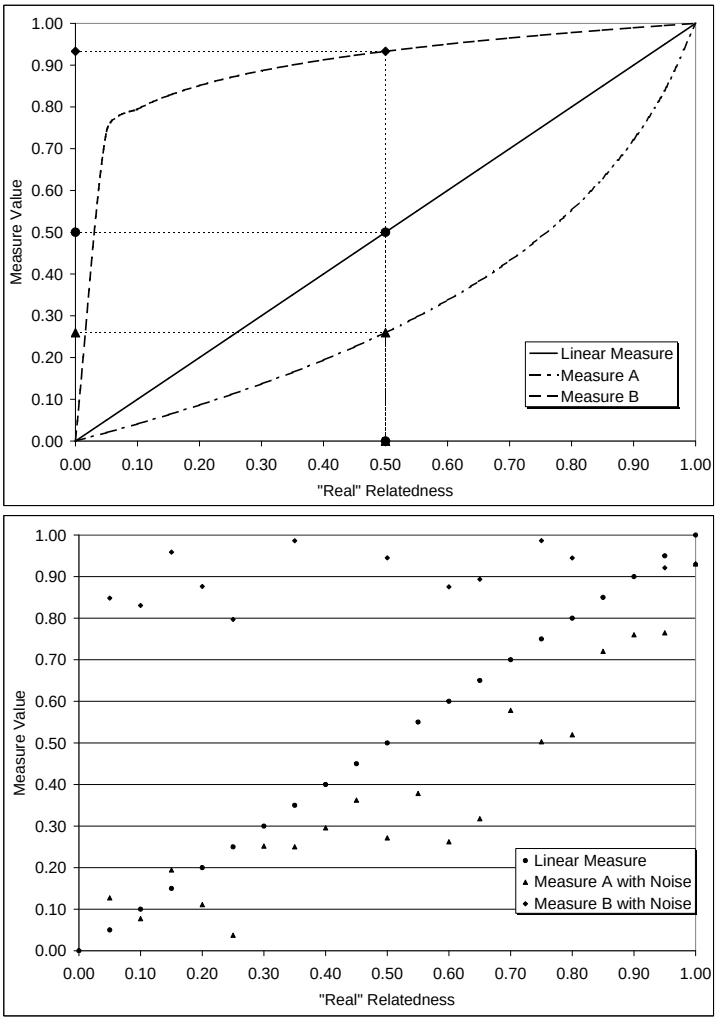

Fig. 6. Idealized (a) and noisy distribution (b) of semantic relatedness values

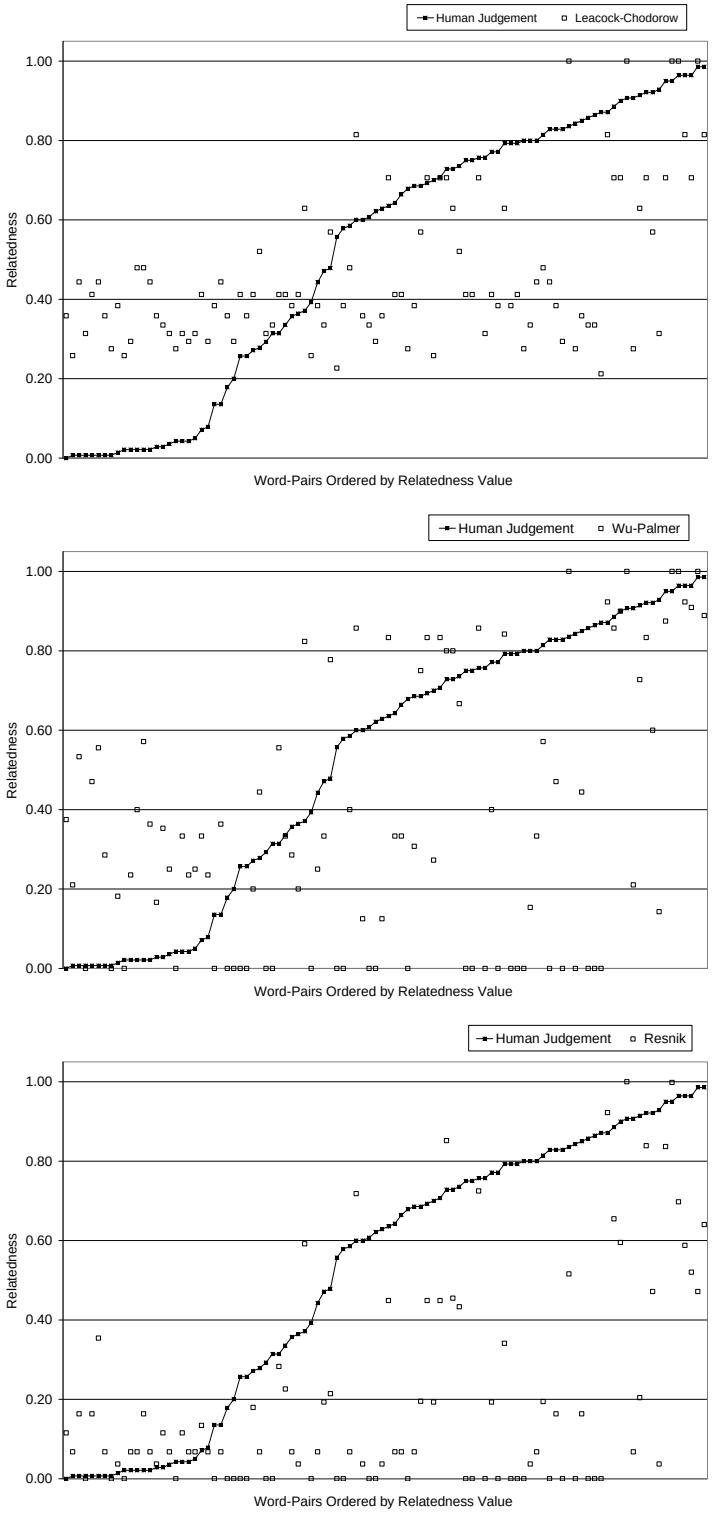

Fig. 7. Leacock-Chodorow (a), Wu-Palmer (b) and Resnik (c) each plotted against human judgment

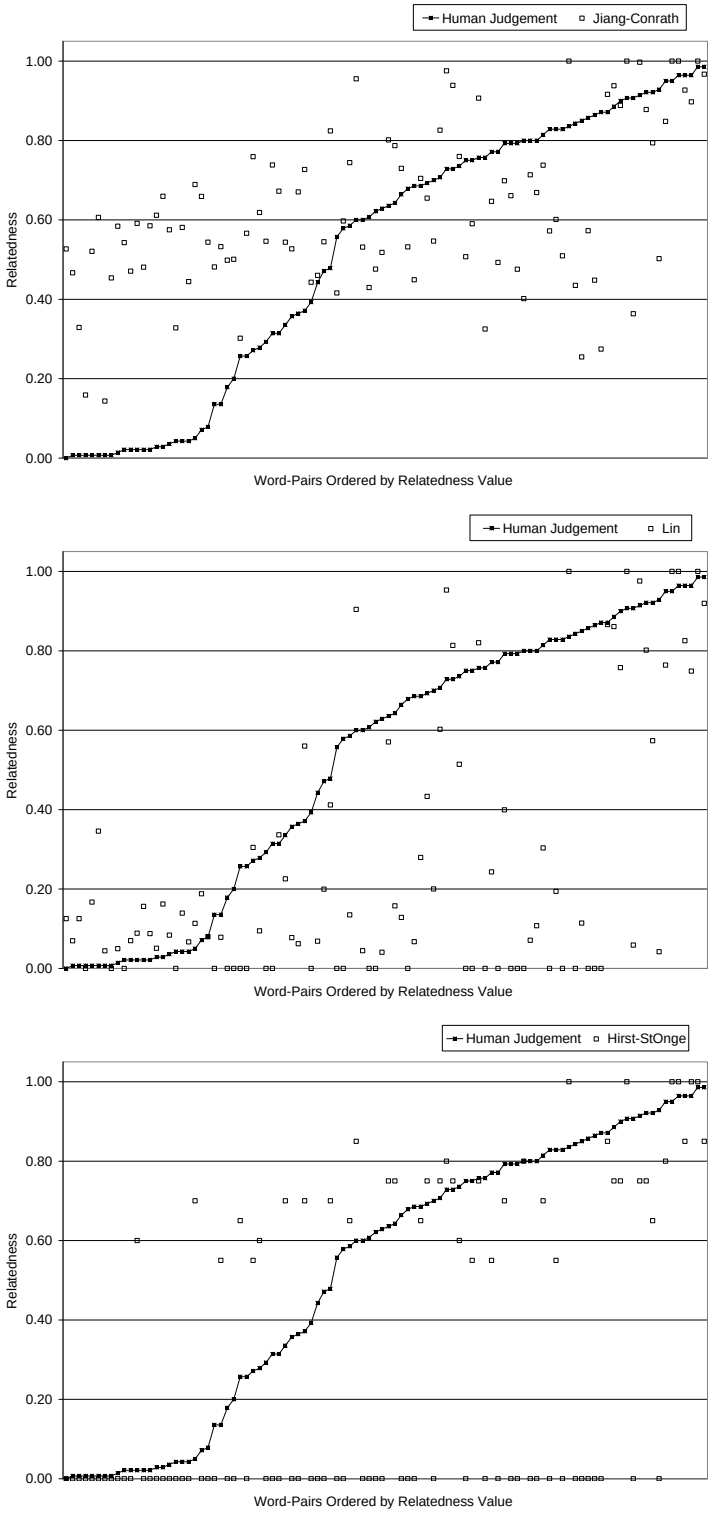

Fig. 8. Jiang-Conrath (a), Lin (b) and Hirst-StOnge (c) each plotted against human judgment

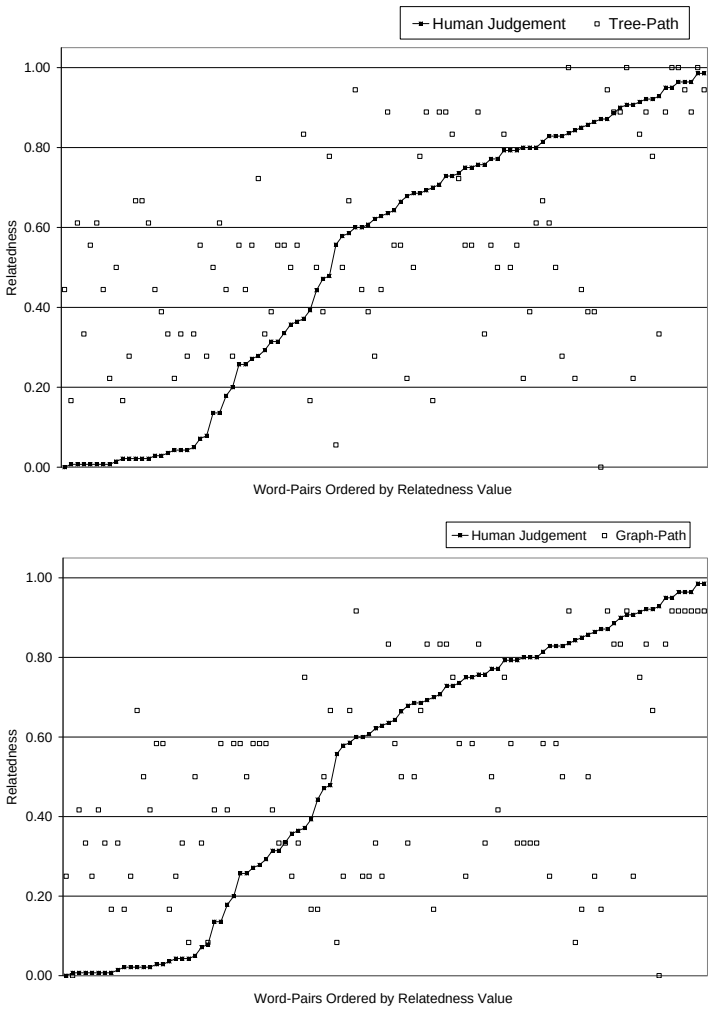

Fig. 9. Tree-Path (a) and Graph-Path (b) each plotted against human judgment

6.2 Comparison of Human Judgment and GermaNet-based Measures

Figures 7a–c, 8a–c and 9a–b show values of the various measures for all word pairs of our human judgment experiment described in Sect. 3.3. Although the inter-annotator agreement in the human judgment experiment is relatively high (correlation: 0.76 +/- 0.04)[11], the correlation between the various measures and the human judgment is relatively low (see Table 7). In addition, the trend functions potentially underlying the (very noisy) graphs in Figures 7a–c, 8a–c and 9a–b are not obvious at all.

Table 7. Correlation coefficients: human judgment vs. relatedness measures

	Graph Path	Tree Path	Wu-Palmer	Leacock-Chodorow
correl. coeff.	0.41	0.42	0.36	0.48
	Hirst-StOnge	Resnik	Jiang-Conrath	Lin
correl. coeff.	0.47	0.44	0.45	0.48

In order to use one of these measures or a combination of them in `GLexi`, we need to determine the best measure(s) and, because a lexical chainer mostly works with classes of relatedness, a function, which maps these values into discrete intervals of relatedness. We question whether a relatedness measure used in a lexical chainer has to be continuous; a continuous value can misleadingly appear to indicate an unrealistic grade of accuracy. Instead, a measure mapping from a list of features, such as relation type, network density or node depth etc., into three classes, such as *not related*, *related* and *strongly related* might be more adequate. The class distribution in our human judgment experiment shown in Fig. 10 confirms this idea. Because of the relatively low correlation between the measure values and the human judgment, the extreme noise in the distribution functions shown in Figures 7a–c, 8a–c and 9a–b, and the fact that interesting features of GermaNet are not yet considered in the calculation of the relatedness values, we assume that none of the measures presented in this paper is in fact appropriate for lexical chaining in German. In our future work we plan to integrate these findings into a Machine Learning based mapping between GermaNet-based features (and word counts, co-occurrence) and discrete classes of relatedness.

7 Evaluation Phase IV – Application-oriented Evaluation

The constraints imposed on our lexical chainer by the application scenario, i.e. the extraction of topic anchors and the topic chaining itself, are as follows: Firstly, we intend to utilize the structure and information about a specific text encoded in the lexical

functions (as mentioned in Sect. 6.1). Although the two are equal with regards to their output (concrete measure values), they differ with respect to their input dimension and type.

[11] The inter-annotator agreement in our study is slightly lower than those reported in the literature for English because we considered systematically and unsystematically related word pairs as well as abstract and tricky nouns.

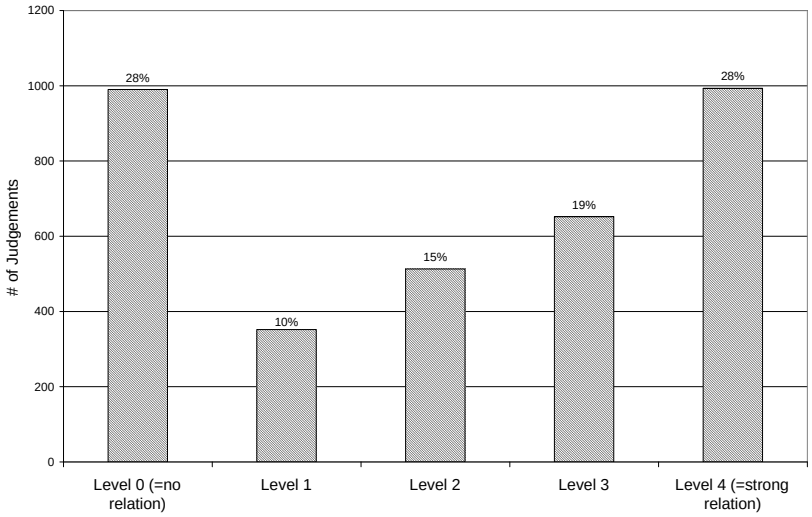

Fig. 10. Distribution of human judgment

chains as input features for the extraction of topic anchors. Especially, the length of a chain, the density and strength of its internal linking structure should be of great importance. Admittedly, additional chaining of independent features could be necessary to ultimately determine the topic anchors of a text passage. Secondly, we plan to use the same algorithms and resources for the construction of both lexical and topic chains. Merely the chaining candidates, i.e. all noun tokens for lexical chaining and exclusively topic anchors for topic chaining, account for the difference between the two types of chaining. However, we assume that for both chaining types a net structure could be superior to linearly organized chains. This kind of structure for a passage of a newspaper article, which we computed on the basis of our lexical chainer, is shown in Fig. 11. The article covers child poverty in German society; accordingly, the essential concepts are `Kind` (Engl. child), `Geld` (Engl. money), `Deutschland` (Engl. Germany), and `Staat` (Engl. state). On the basis of, among other things, edge density and frequency, we calculated the most relevant words (especially, `Kind`, `Geld`, `Deutschland`, and `Staat`), which we then accordingly highlighted in the graph shown in Fig. 11. Finally, the parameter settings, which we found to be reasonable on the basis of the evaluation phases I–III, need to be integrated with the constraints imposed on our lexical chainer by our application in our future work.

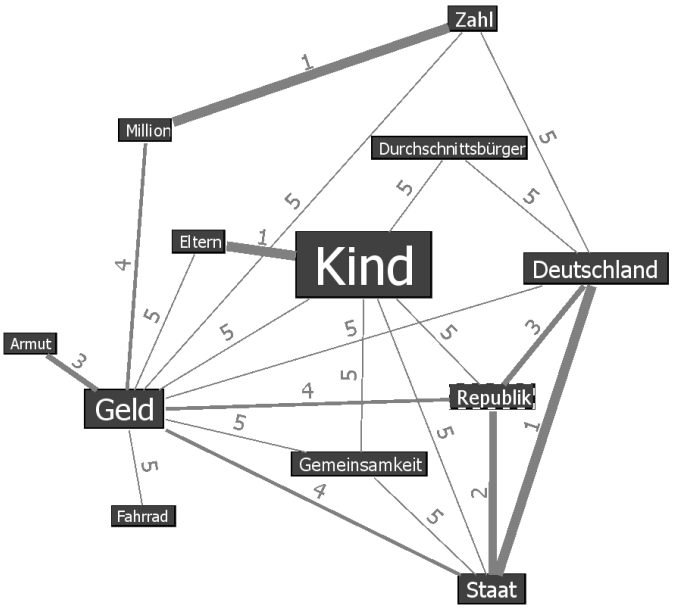

Fig. 11. Input for topic chaining: net structure-based lexical chaining example

8 Conclusions and Future Work

We explored the various components and aspects of lexical chaining for German corpora of technical and academic documents. We presented a detailed evaluation procedure and discussed the performance of our chaining system with respect to these aspects. We could show that preprocessing plays a major role due to of the complex morphology in German and furthermore that technical terminology and proper names are of great importance. Additionally, we discussed the performance of a simple chaining-based word sense disambiguation and outlined a method to enhance this aspect. We also presented a human judgment experiment which was conducted in order to evaluate the various semantic relatedness measures for GermaNet. We were able to show that it is thus far very difficult to determine the function mapping between the measure values and relatedness classes.

We now plan to continue this work on four levels: Firstly, we hope to further improve the preprocessing; i.e. we plan to enhance the compound analysis and the basic NER system. In addition, we intend to integrate components for the handling of abbreviations and technical terminology. Secondly, we aim to develop a sophisticated chaining-based disambiguation methodology which incorporates the idea of meta-chains and other potentially useful features. Thirdly, we plan to investigate alternative relatedness measures, especially Machine Learning based approaches, which map between sets of features and discrete classes of relatedness. Finally, we intend to further explore our

lexical chainer with respect to topic chaining and thus to evaluate our chainer in an application oriented manner.

References

Regina Barzilay and Michael Elhadad. 1997. Using lexical chains for text summarization. In *Proc. of the Intelligent Scalable Text Summarization Workshop (ISTS'97)*.

Beata Beigman Klebanov. 2005. Using readers to identify lexical cohesive structures in texts. In *Proc. of ACL Student Research Workshop (ACL2005)*.

Michael Beißwenger and Sandra Wellinghoff. 2006. Inhalt und Zusammensetzung des Fachtextkorpus. Technical report, University of Dortmund, Germany.

Michael Beißwenger, Angelika Storrer, and Maren Runte. 2003. Modellierung eines Terminologienetzes für das automatische Linking auf der Grundlage von WordNet. In *LDV-Forum, 19 (1/2) (Special issue on GermaNet applications, edited by Claudia Kunze, Lothar Lemnitzer, Andreas Wagner)*.

Alexander Budanitsky and Graeme Hirst. 2001. Semantic distance in wordnet: An experimental, application-oriented evaluation of five measures. In *Workshop on WordNet and Other Lexical Resources at NAACL-2000*.

Joe Carthy. 2004. Lexical chains versus keywords for topic tracking. In *Computational Linguistics and Intelligent Text Processing*, Lecture Notes in Computer Science. Springer.

Christiane Fellbaum, editor. 1998. *WordNet. An Electronic Lexical Database*. The MIT Press.

Stephen J. Green. 1999. Building hypertext links by computing semantic similarity. *IEEE Transactions on Knowledge and Data Engineering*, 11(5).

Iryna Gurevych and Hendrik Niederlich. 2005. Computing semantic relatedness in german with revised information content metrics. In *Proc. of OntoLex 2005 - Ontologies and Lexical Resources, IJCNLP 05 Workshop*.

Iryna Gurevych. 2005. Using the structure of a conceptual network in computing semantic relatedness. In *Proc. of the IJCNLP 2005*.

M. A. K. Halliday and Ruqaiya Hasan. 1976. *Cohesion in English*. Longman, London.

Graeme Hirst and David St-Onge. 1998. Lexical chains as representation of context for the detection and correction malapropisms. In Christiane Fellbaum, editor, *WordNet: An electronic lexical database*.

Jay J. Jiang and David W. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. *Proc. of the International Conference on Research in Computational Linguisics*.

Claudia Kunze, Lothar Lemnitzer, Harald Lüngen, and Angelika Storrer. in this volume. Towards an integrated owl model for domain-specific and general language wordnets.

Claudia Leacock and Martin Chodorow. 1998. Combining local context and wordnet similarity for word sense identification. In Christiane Fellbaum, editor, *WordNet: An electronic lexical database*.

Lothar Lemnitzer and Claudia Kunze. 2002a. Adapting germanet for the web. In *Proc. of the 1st Global Wordnet Conference (GWC2002)*.

Lothar Lemnitzer and Claudia Kunze. 2002b. Germanet - representation, visualization, application. In *Proc. of the Language Resources and Evaluation Conference (LREC2002)*.

Eva Anna Lenz and Harald Lüngen. 2004. Annotationsschicht: Logische Dokumentstruktur. Technical report, University of Dortmund, Germany.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proc. of the 15th International Conference on Machine Learning*.

Alexander Mehler. 2005. Lexical chaining as a source of text chaining. In *Proc. of the 1st Computational Systemic Functional Grammar Conference, Sydney*.

George A. Miller and Walter G. Charles. 1991. Contextual correlates of semantic similiarity. *Language and Cognitive Processes*, 6(1).

Jane Morris and Grame Hirst. 1991. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational linguistics*, 17(1).

Jane Morris and Grame Hirst. 2004. Non-classical lexical semantic relations. In *Proc. of HLT-NAACL Workshop on Computational Lexical Semantics*.

Jane Morris and Grame Hirst. 2005. The subjectivity of lexical cohesion in text. In J. C. Chanahan, C. Qu, and J. Wiebe, editors, *Computing attitude and affect in text*. Springer.

Adrian Novischi and Dan Moldovan. 2006. Question answering with lexical chains propagating verb arguments. In *Proc. of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*.

Philip Resnik. 1995. Using information content to evaluate semantic similarity in a taxonomy. In *Proc. of the IJCAI 1995*.

Herbert Rubenstein and John B. Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8(10).

Grogory H. Silber and Kathleen F. McCoy. 2002. Efficiently computed lexical chains as an intermediate representation for automatic text summarization. *Computational Linguistics*, 28(4).

Maik Stührenberg, Daniela Goecke, Nils Diewald, Alexander Mehler, and Irene Cramer. 2007. Web-based annotation of anaphoric relations and lexical chains. In *Proc. of the Linguistic Annotation Workshop, ACL 2007*.

Elke Teich and Peter Fankhauser. 2004. Wordnet for lexical cohesion analysis. In *Proc. of the 2nd Global WordNet Conference (GWC2004)*.

Zhibiao Wu and Martha Palmer. 1994. Verb semantics and lexical selection. In *Proc. of the 32nd Annual Meeting of the Association for Computational Linguistics*.

Torsten Zesch and Iryna Gurevych. 2006. Automatically creating datasets for measures of semantic relatedness. In *Proc. of the Workshop on Linguistic Distances (ACL 2006)*.