Volume 23 – Number 2 – 2008 – ISSN 0175-1336

JLCL

Journal for Language Technology
and Computational Linguistics

Lexical-Semantic Resources in Automated Discourse Analysis

Edited by
Harald Lüngen, Alexander Mehler and Angelika Storrer

GSCL Gesellschaft für Sprachtechnologie und Computerlinguistik www.gscl.info

Lexical-Semantic Resources in Automated Discourse Analysis

Impressum

JLCL
ISSN 0175-1336
Volume 23 (2) – 2008

Journal for Language Technology and Computational Linguistics – offizielles Organ der GSCL

Herausgeber

Gesellschaft für Sprachtechnologie und Computerlinguistik (GSCL)
Dr. Alexander Mehler, Goethe-Universität Frankfurt am Main, *Mehler@em.uni-frankfurt.de*
Prof. Dr. Christian Wolff, Universität Regensburg *christian.wolff@sprachlit.uni-regensburg.de*

Anschrift der Redaktion

Prof. Dr. Christian Wolff,
Universität Regensburg
Institut für Medien-, Informations- und Kulturwissenschaft
D-93040 Regensburg

Wissenschaftlicher Beirat

Vorstand, Beirat und Arbeitskreisleiter der GSCL
http://www.gscl.info/vorstand.html,
http://www.gscl.info/

Erscheinungsweise

2 Hefte im Jahr, halbjährlich zum 31. Mai und 31. Oktober. Preprints und redaktionelle Planungen sind über die Website der GSCL einsehbar (*http://www.gscl.info*).

Einreichung von Beiträgen

Eingesandte Fachbeiträge werden vor Veröffentlichung von mindestens zwei ReferentInnen begutachtet. Manuskripte sollten deshalb möglichst frühzeitig eingereicht werden und bei Annahme zur Veröffentlichung in jedem Fall elektronisch und zusätzlich auf Papier übermittelt werden. Die namentlich gezeichneten Beiträge geben ausschließlich die Meinung der AutorInnen wieder. Einreichungen sind an die Herausgeber zu übermitteln.

Bezugsbedingungen

Für Mitglieder der GSCL ist der Bezugspreis des JLCL im Jahresbeitrag mit eingeschlossen. Jahresabonnements können zum Preis von 25,- € (inkl. Versand), Einzelexemplare zum Preis von 15,- € (zzgl. Versandkosten) bei der Redaktion bestellt werden.

Satz und Druck

Boris Kaiser, Bielefeld, mit *LaTeX* (*pdfeTeX* / *MiKTeX*) und *Adobe InDesign CS*
Druck: Druck TEAM KG, Regensburg

Harald Lüngen, Alexander Mehler, Angelika Storrer

Editorial

Following the special editions 22(2) and 23(1) on *Foundations of ontologies in text-technology*, this is the third volume of the LDV-Forum to originate from activities of the research unit 437 *Text-technological modelling of information*, funded by the German Research Foundation (DFG) in its second phase 2005-2009. One of the research goals shared by four out of five subprojects within the group was the automated analysis of different types of discourse relations and the construction and evaluation of domain ontologies and lexical-semantic wordnets as knowledge sources in this task.

The subproject HyTex - *Text-Grammatical Foundations for the (Semi)-Automated Text-to-Hypertext Conversion* (Principal Investigator: Angelika Storrer) was concerned with the identification of thematic development structures in specialised texts for the purpose of hypertextualisation. The goal of the subproject *Indogram - Induction of document grammars for the Representation of Logical Hypertextual Document Structures* (Principal Investigator: Alexander Mehler) was to research methods of learning document and content structures from very large corpora of hypertext (web) documents. Finally, the subproject *SemDok - Generic Document Structures in Linearly Organised Texts* (Principal Investigator: Henning Lobin) dealt with building a text parser in the framework of Rhetorical Structure Theory for the complex text type of scientific research articles. As a joint intitiative of these three projects, a workshop entitled *Ontologies and Semantic Lexica in Automated Discourse Analysis* was held in conjunction with the *Arbeitskreis Korpuslinguistik* of the GLDV at the GLDV Frühjahrstagung in April 2007 in Tübingen.

The workshop included a most inspiring invited talk by Manfred Stede, entitled "Gewusst wie - Ontologien und Textkohärenz", which presented an insightful overview of the history of automatic text understanding systems from early knowledge-based systems within the discipline of Artificial Intelligence in the 1970s and 1980s to the rise of statistical methods in the 1990s and to more recent hybrid approaches and his own text-technological, multi-level analysis approach to text processing. A concise version of this historical overview is now contained as a part of Stede's contribution to the present volume.

Altogether, four of the five contributions to the present volume are paper versions of talks held at the workshop, namely the ones by Manfred Stede, Caroline Sporleder, Bärenfänger et al. and Diewald et al. The fifth article by Cramer et al. was additionally reviewed and included due to its immediate relevance for the field of lexical-semantic ressources in automated discourse analysis. Each paper was reviewed for the LDV-Forum by two external reviewers; as is customary, reviewers for the first round were chosen by the guest editors, and reviewers for the second round were chosen by the regular editors of the LDV-Forum.

Discourse analysis in the title of this volume, and, relatedly, *discourse structure* and *discourse relations* are used as cover terms for various types of relational structuring of text beyond the domain of sentences. Several levels of discourse structure can be identified on account of the types of relations used and the linguistic units involved in them. Let us briefly review the most important levels of discourse structure and current theoretical and practical approaches to their (automated) identification in text.

The first one is the level of coreference, or anaphora. Anaphora is a cohesive phenomenon that occurs intra-sententially as well as inter-sententially. Anaphoric relations are usually described to hold between discourse entities that are elements of the semantic model of the text world, or alternatively, between the linguistic units that are used to express them. Current anaphora resolution systems such as described in Vieira (2000) or Stuckardt (2005) use large amounts of annotated training corpora. Diewald et al.'s (pp. 74–92) contribution to this volume describes a novel, web-based multi-user annotation tool for semantic relations which can be used to produce corpora for a specialised task like anaphora resolution. In Bärenfänger et al. (pp. 49–72), an ontology of anaphoric relation types is introduced, and the interrelationship between anaphora, rhetorical relations, and thematic development is examined in a corpus study based on linguistic annotations on multiple levels.

On another level of discourse structure, which may be called lexical cohesion after Halliday and Hasan (1976), the content words occurring in a text are grouped into lexical chains based on lexical-semantic relations holding between them. Lexical chaining is a computational linguistic application first introduced by Morris and Hirst (1991). In their original algorithm, chains were derived by means of looking up lexical-semantic relations between content words in Roget's Thesaurus. More recent approaches to lexical chaining use wordnets as a knowledge source (Hirst and St-Onge, 1998), or establish semantic relatedness using terminologies or social ontologies such as Wikipedia (Mehler 2009, Waltinger et al., 2008). The third article of this volume (Cramer et al. pp. 34–47) describes experiences with the development of the lexical chainer GLexi, which derives semantic distances from GermaNet and was tested on the HyTex corpus of German specialised texts. GLexi is ultimately supposed to function as a component in the generation of topic views as an automated hypertextualisation strategy applied to a given linear text. Topic views can be regarded as an approximation to a representation of thematic development in a text.

Many researchers use the term *discourse structure* exclusively as a label for the system of coherence relations between text segments of different size, usually with propositional content. Examples of discourse relations of this type are the causal, the contrastive, or the elaboration relation. A number of discourse theories aim at describing the admissible structures of text-type-independent discourse coherence relations, notably SDRT (Asher and Lascarides, 2003), RST (Mann and Taboada, 2006), or the ULDM (Polanyi et al., 2003). Discourse coherence relations are frequently associated with lexical items called *discourse connectives*, but at the same time, many coherence relation instances in

a text lack overt cues. The contribution by Sporleder (p. 20–32) in this volume presents an evaluation of machine learning models in which lexical-semantic relations from the Princeton WordNet are used to disambiguate discourse coherence relations from SDRT that lack overt or unambiguous discourse markers. The contribution by Bärenfänger et al. (p. 49–72) explores the question how types of anaphoric relations annotated in a corpus of scientific articles can help identify instances of the RST elaboration relation.

We would like to thank both the review board for the extended abstracts submitted for the Tübingen workshop as well the reviewers of the paper versions. Without their support we would not have accomplished another edition of the LDV-Forum of such high calibre: Irene Cramer, Marcus Egg, Christiane Fellbaum, Iryna Gurevych, Anke Holler, Kai-Uwe Kühnberger, Peter Kühnlein, Lothar Lemnitzer, Henning Lobin, Vivian Raithel, Georg Rehm, Roman Schneider, Bernhard Schröder, Manfred Stede and Christian Wolff. Furthermore, we would like to thank the editors of the LDV-Forum, Alexander Mehler and Christian Wolff and their team for their support and advice. We hope that the readers of the LDV-Forum will find the included papers as interesting and illuminating as we did.

December 2008

Harald Lüngen
Alexander Mehler
Angelika Storrer

References

Literatur

Asher, N. and Lascarides, A. (2003). *Logics of Conversation*. Cambridge University Press, Cambridge, UK.

Halliday, M. and Hasan, R. (1976). *Cohesion in English*. English Language Series. Long man, London, 5 edition.

Hirst, G. and St-Onge, D. (1998). Lexical chains as representation of context for the detection and correction malapropisms. In Fellbaum, C., editor, *WordNet: An electronic lexical database.*

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.

Polanyi, L., van den Berg, M., and Ahn, D. (2003). Discourse structure and sentential information structure. *Journal of Logic, Language and Information*, 12:337–350.

Stuckardt, R. (2005). A machine learning approach to preference strategies for anaphor resolution. In Branco, A., McEnery, T., and Mitkov, R., editors, *Anaphora Processing: Linguistic, Cognitive, and Computational Modelling*. John Benjamins.

Taboada, Maite and Mann, William C. (2006). Rhetorical Structure Theory: Looking back and moving ahead. *Discourse Studies*, 8(3):423–459.

Vieira, R. u. M. P. (2000). An empirically-based system for processing definite descriptions. *Computational Linguistics*, 26(4):525–579.

Waltinger, U., Mehler, A., and Stührenberg, M. (2008). An integrated model of lexical chaining: Applications, resources and its format. *In KONVENS 2008 - Ergänzungsband Textressourcen und lexikalisches Wissen*, pages 59–70, Berlin.

Alexander Mehler (2009): A Quantitative Graph Model of Social Ontologies by Example of Wikipedia. In: Alexander Mehler, Serge Sharoff and Marina Santini (eds.): *A Quantitative Graph Model of Social Ontologies by Example of Wikipedia. Genres on the Web: Computational Models and Empirical Studies*, Forthcoming.

Inhaltsverzeichnis

Journal for Language Technology and Computational Linguistics - Volume 23(2) - 2008
Lexical-Semantic Resources in Automated Discourse Analysis

Manfred Stede

Local coherence analysis in a multi-level approach to automatic text analysis

We characterize a text-technological approach to text analysis as combination of a multi-level representation framework and XML-based document processing techniques. The main advantages of such an approach are the chance to flexibly combine modules for constructing different applications, and the overall robustness resulting from the operational principle of higher-level modules combining the — possibly partial — results of lower-level ones. We illustrate the approach with the specific task of local coherence analysis, i.e. the computation of coherence relations between text spans.

1 Introduction

What does it mean to analyze or — more ambitiously — to understand a text? Over the years, Artificial Intelligence and Computational Linguistics have responded in quite different ways to this question. The present paper argues in favour of a text-technologically-inspired multi-level approach to automatic text analysis: *Text technology* emphasizes the utility of XML-based document processing techniques (see Lobin (2000)), while the *multi-level* conception views text analysis as the systematic composition of distinct levels of information that can be produced by independent analysis tools. By bringing these two realms together, we aim at designing robust systems that can be easily configured for dealing with different kinds of text and perform different tasks (which usually share a number of generic subtasks such as tagging, noun phrase chunking, etc.). When levels are created independently of one another, the analysis tools might very well produce better or worse results for different portions of the text. Looking then across all levels of analysis, we can end up with more or less information for different segments — depending on how difficult the text is, and on how good the tools are. The result is what had been envisaged by Hirst and Ryan (1992) as a "mixed-depth representation" of text content. Utilizing contemporary text-technological approaches, this can be achieved by a highly distributed analysis approach much better than with "holistic" analysis schemes that had been *en vogue* when Hirst and Ryan had put forward the idea.

After outlining this general approach (Sections 2 to 4), the second half of the paper (Section 5) will illustrate the idea by focusing on one particular subtask of text understanding: *local coherence analysis*, i.e., the inferring of semantic or pragmatic relations holding between text segments. As long as "deep semantics" and knowledge processing are absent, the most useful information for such an analysis module comes from the *connectives*: closed-class lexical items that express, more or less specifically, the semantic or pragmatic relationship between text segments (or, more precisely, between their interpretations).

After illustrating the task of local coherence analysis with a simple example, we will enumerate the problems that connectives can create for text analysis, and then sketch an approach to automatic local coherence analysis that is embedded in the multi-level framework.

2 Looking back (1): Text Understanding in Artificial Intelligence

In the 1970s, automatic text understanding was one of the central goals of the flourishing discipline of Artificial Intelligence, which at the time aimed at reproducing human cognitive behaviour with machines employing symbolic representations and inference mechanisms. For a cognitive agent, understanding a text was largely conceived as aligning the text with the prior knowledge of that agent – a mechanism that was quite explicitly formulated in the *script* representations and alignment procedures in the tradition of Schank and Riesbeck (1981). As an extension of the popular paradigm of encoding static, factual knowledge with *semantic networks*, scripts were meant to represent an agent's procedural knowledge about stereotypical events. The best-known example is the *restaurant* script that encodes the steps of entering a restaurant; choosing and ordering food; eating; paying the bill, and leaving. One of the early programs, SAM, was able to match simple English stories against this generic script and thereby to "understand" a particular story about somebody eating something in some restaurant. It is important to notice that SAM did not perform anything like a syntactic analysis but directly matched surface patterns of English words against a meaning representation in the framework of *conceptual dependency* theory, an approach that aimed to represent word meaning (and in particular that of verbs) by decomposing it into semantic nets consisting of a set of *primitives* and relations between them. Consequently, programs like SAM always operated on carefully hand-crafted sample texts to which the (equally carefully handcrafted) conceptual dependency patterns would fit.

An important step forward from such toy settings was the FRUMP program by DeJong (1982). FRUMP in fact took news messages as input, which were — in contrast to SAM — not required to belong to a single small domain. Instead, FRUMP first inspected the text for possible topics and then actively selected the script which it surmised to be most suitable. For doing that, FRUMP drew on a set of 40 precoded scripts, which represented typical flows of reports on certain types of news. This difference in coverage made FRUMP much more impressive than SAM, but at the same time, the work in retrospect made it clear that the overall approach of pre-coding "story scripts" was a dead-end: In order to extend the program to further coverage, one would have to write many more scripts; moreover, the approach of "semantics-only" pattern matching was not very tolerant to mild deviations in the story and/or its linguistic formulation. Clearly, some general notion of paraphrasing was needed in order to detect that a great many linguistic variants could in effect report on the exact same event.

In the 1980s, The German LILOG project (Herzog and Rollinger, 1991) set out to avoid such fallacies by adopting a highly modular approach that clearly distinguished between knowledge sources such as syntax, lexical semantics, sentence meaning, and pre-

coded background knowledge. It performed a thorough syntactic and semantic analysis and linked the resulting meaning representations of sentences with a knowledge base encoding domain and world knowledge, in order to account for "text meaning" beyond the sentence. All modules were developed with intensively-researched formalisms, and much care was taken in devising the scheme of interaction between the modules. In this way, LILOG produced many important results on linguistic representation and processing, and it also lead to a working implementation — but this was able to analyze hardly more than one sample text that had been chosen beforehand as illustration of the phenomena that had to be tackled. In essence, the problem was that each module relied on the completeness and correctness on the output of those modules that preceded it in the processing pipeline, and since all modules obviously had their individual weaknesses and gaps of coverage, the *overall* coverage of the system was low. Thus, while adopting a much more principled and linguistically-minded approach than the aforementioned works in the conceptual dependency tradition, LILOG also ran into the problem that breadth of coverage was extremely difficult to achieve: the problem that has become well-known as that of *robustness*.

3 Looking back (2): Statistical Methods for Text-oriented Applications

What is today called the "statistical turn" of Computational Linguistics in the early 1990s was quite probably a consequence of two distinct developments: on the one hand the growing frustration with AI-style systems that sometimes produced interesting toy solutions but invariably failed to "scale up"; on the other hand the impressive results of acoustic speech recognition, which surfaced in the late 1980s and which were entirely based on statistical methods, with no linguistic representations involved.

Now, emphasis shifted to research that clearly aimed at practical applications, that was able to process realistic data, and that followed strict methodologies of evaluating one's work in quantitative manners. Accordingly, Computational Linguistics grew more into an engineering-like discipline, and towards the end of the decade, the term *Language Technology* became widely accepted as a label for efforts to bring language-processing applications into the "real world".

For text-oriented research, a significant milestone was produced with the *message understanding conference* (MUC) competitions sponsored by DARPA in the 1990s (Grishman and Sundheim, 1996). The goal was to identify specific pieces of information from authentic texts (news messages) that belonged to a particular genre, such as terrorist attacks. In this case, the software should be able to extract, e.g., the type of attack, its target, the purported agent(s), date and location of the attack. MUC was organized as an open competition for interested research teams; performance was evaluated by clear quantitative criteria, and the conferences generated both much attention and considerable progress. Quite soon, the evaluations were run not only with respect to overall performance but also for various subtasks involved, such as reference resolution. Hence, a team could also demonstrate its strength by focusing their attention on specific NLP tasks, which in turn lead to more sophisticated methods for handling such tasks. Subsequent events that

were run in a similar manner were the *Document Understanding Conference* (DUC) and the *Text Retrieval Conference* (TREC). Here, the original task of information extraction was extended to challenges on machine translation, cross-linguistic information retrieval, text summarization, or question answering.

For all these purposes, many approaches were developed, the vast majority of which was based almost exclusively on statistical methods, i.e., by training automatic classifiers on labelled data. From the applications-oriented perspective, this definitely lead to success: Open-domain question-answering, for instance, nowadays can be done to an extent that just a few years ago few would have thought to be possible. On the other hand, since the statistical models are largely intransparent "black boxes", and furthermore, the MUC/DUC/TREC modules have usually been extensively tailored to the particular domain and genre in question, not too many generalizable insights into the principles of document structure and text analysis have been gathered in this way. In short, while task performance increased significantly, the interest in (text-)linguistic insights has simultaneously shrunk.[1]

4 A Text-Technological Approach to Text Analysis

On the basis of the (necessarily subjective) recap of the history of automatic text analysis given above, we will now sketch the idea of a *text-technological* perspective, which tries to avoid the pitfalls of AI-inspired approaches (Section 1) and at the same time tries not to be as narrow-minded as many purely-statistical approaches (Section 2). Instead, we aim at viewing the task of text analysis as a matter of XML-based document processing steps that can work together in a manner as modular and flexible as possible.

4.1 Analysis on Multiple Levels

As is well-known, most text analysis applications nowadays share a certain base set of processing steps, such as identifying the logical document structure, performing part-of-speech tagging and possibly some sort of phrase chunking; on top of these results, more abstract and possibly more application-specific modules can be run. The idea of *multi-level analysis* is to implement this approach in a highly systematic way, i.e., by extending the text document step by step with linguistic and other information, which in turn can be used by further analysis steps to add even more information.

These additional modules may compute "classical" linguistic information such as syntactic structures, be it in the form of dependency trees or constituent structures. One task that needs to be performed in addition to standard parsing is *named-entity recognition*; while these entities usually correspond to linguistic constituents (noun phrases), they

[1] Granted, we are simplifying here: There has also been work on the listed applications with more linguistics or knowledge processing involved; see, for example, Hovy et al. (2002) for knowledge-based question answering. Nonetheless, it is certainly fair to say that in Language Technology, purely statistical work has a clear majority. One side-effect, demonstrated with a quantitative analysis by Reiter (2007), is the rapidly shrinking amount of research that methodologically is situated on the border to Psychology or Cognitive Science.

can be quite complex and are, by definition, not covered by the standard lexicon used by a parser. Then, however, there are several tasks in text analysis that do *not* correspond to levels of standard linguistic analysis. On the contrary, some of the issues are even responsible for the robustness problems of early syntactic parsers. For example, a module for identifying time and date expressions performs a very useful job for any application that needs to track the temporal unfolding of events; and precisely this class of time and date expressions was one prominent cause of disturbance for syntactic parsing based on symbolic grammars in the early 1980s (see also Stede (1992)). A well-known approach to annotating temporal expressions in English is the TimeML framework[2], which also extends to capturing the linguistic marking of event structure. For German, an inventory of temporal expressions has for instance been compiled as part of the *Verbmobil* project by Endriss et al. (1998). Building on this theoretical proposal, our implementation of a time/date analyzer (Luft, 2007) operates immediately on the tokenized text and contributes a new level of information independent of syntactic analysis. In subsequent analysis steps, the temporal expressions can be interpreted (i.e., projected onto the calendar) and then support any application that can profit from this information.

There are quite a few other examples of "non-standard" levels for representing information found in texts. For the applications of opinion mining or question-answering, for instance, it is important to gather from the text whose viewpoint is expressed in a certain passage: A particular piece of information might not be a statement of the author but in a more or less complicated fashion be attributed to someone else. The most obvious step here is to identify quoted speech and the speaker whom it is attributed to. With indirect speech and the many ways of expressing it in text, this becomes more complicated. The "ultimate" solution would go as far as tracking the point-of-view as it develops in the text, demonstrated with a rather complex algorithm for the narrative text type by Wiebe (1994). Our current implementation of an 'attibution recognizer' is much more modest and merely tries to recognize quoted as well as indirect speech. It identifies the former on the level of plain text tokens and the latter on the output of a dependency parser, using search patterns formed with communication verbs. Again, it contributes to the pool of analyses a new layer that marks attributed content and links it to the reported source of that information. This level can in turn be used by a generic rhetorical parser — see for instance Marcu (2000), who had pointed out the problems that attributed content poses for rhetorical parsing.

To give a third and final example, for many applications it is important to know whether a certain passage in a text is presented by the author as "objective" information or marked as a subjective statement, either on behalf of the author herself or on behalf of somebody mentioned in the text — in which case this task links up to that of viewpoint identification mentioned above. To make this more concrete, consider again the application of question-answering and suppose the user asked "When was Barack Obama born?" In order to answer this reliably, a system should not just match the keywords but be able to distinguish the follwing text passages:

[2] http://www.timeml.org

1. *Barack Hussein Obama, Jr. ([...]; born August 4, 1961) is the junior United States Senator from Illinois.* (en.wikipedia.org, March 3rd, 2008)

2. *Some Republicans claimed that Obama was born in 1965.* (fictitous)

3. *Obama was probably born on the Fourth of July in 1961.* (fictitous)

While (1) purports to "state the facts", (2) explicitly attributes the information on the year of birth to a third party (pardon the pun); with (3), finally, the author indicates that he is not quite sure whether the information he provides is actually correct. We use the term "subjectivity identification" for the overall task of noticing expressions of epistemic stance (as in (3)) and of discovering opinion, i.e., finding out whether the author merely presents a fact or indicates that she likes or dislikes a particular state of affairs.

There are several other useful examples of such intermediate processing tasks that are relevant not to all applications of text understanding but to many of them. Accordingly, tasks like these should not be implemented from scratch with any new application or research project but treated as independent modules that can contribute their share to the bigger task of making sense of a text. Thus, in a multi-level framework, "making sense" ist not an "all-or-nothing" endeavour; instead, the idea is to employ a set of specialized modules, some of which will run independently while some will build on the results of others, so that in the end all information gathered about a text can be read off a stack of separate analyses — and combining information from different stacks can in turn yield more information. Robustness arises when the modules do not break in case some information on "lower" levels is absent but produce either some underspecified representation, or gracefully move on to the next sentence, leaving a gap in the that particular analysis. Obviously, the resulting mixed-depth representations (cf. Section 1) are useful only to the extent that the higher-level modules can actually make use of them, by evaluating underspecifications or by ignoring gaps.

4.2 Processing Architectures

When it comes to actually implementing a multi-level approach, the text-technological perspective adds to the conceptual framework the technical notion of XML-based document processing, which plays a central role in enabling the interoperability of the various analysis modules. The principal challenge is to channel the same source document – possibly enriched with some annotations already – through various black-box analysis modules and to align the output of those modules so that all annotations are in fact correctly assigned to the intended spans of text. An early and successful implementation of the idea was the GATE system (Cunningham et al., 2002), which comes with a number of pre-installed components for analyzing English documents, but turns out somewhat cumbersome to use when "own" modules are to be added to the analysis pipeline. The more recent UIMA architecture (Götz and Suhre, 2004) by IBM follows similar goals as GATE but is more ambitious, targeting large-volume data processing of not only textual but also speech and video data. UIMA defines the XML interface that components must

adhere to, and then manages these components and the data flow between them, which essentially amounts to a pipeline architecture. Components need to be written in Java or C++, or wrappers need to be provided.

Prior to the release of UIMA, several smaller-scale approaches to XML-based document processing frameworks have been developed. The system developed in the 'Sekimo' project (Goecke et al., 2003) maps the information from a set of annotation layers to a Prolog fact base, upon which further computation can be done. For example, Lüngen et al. (2006) use this approach to build a chart parser that analyzes coherence relations between text spans (see next section). In the architecture developed at Potsdam University, a generic standoff-XML representation format called PAULA (Potsdamer AUstauschformat für Linguistische Annotation; Dipper (2005)) has been defined, along with conversion scripts that map the output of various annotation tools and analysis modules to PAULA (Chiarcos et al., 2008). In this approach, the aim is to use the same architecture both for the scenario of manual annotation and for automatic processing. Regarding the first goal, the linguistic database ANNIS serves to integrate multiple annotations of the same text (e.g., syntax, coreference, focus/background structure) and enables querying across levels, as well as some statistical analyses. ANNIS serves as the central repository for data collected within the *Sonderforschungsbereich* 632 'Information Structure' at Potsdam University and Humboldt-University Berlin, but is also available to external researchers.[3] The first version of ANNIS relied on a representation of the information as Java objects in main memory, which are designed to specifically support effective visualization and querying. At present, version 2 is under development, which adds an interface to a relational database management system (PostgreSQL) so that larger amounts of data can be handled.

Besides the scenario of manual annotation of linguistic data on multiple levels, the PAULA framework is also used in applications of automatic text processing. Our implementation of a 'Modular Text analysis System' (MOTS) currently integrates about a dozen different modules (both freely available tools and modules developed by ourselves) equipped with wrappers that ensure their compatibility with PAULA. One important aspect of wrapping is to ensure consistent tokenization: All analyses in the various layers (often transitively) refer to a unique 'token' layer, which in turn refers to the source document. Hence, for analysis modules that come with their own built-in tokenizer, a workaround must be defined as part of the wrapping. Our pilot application was a text summarizer built in the SUMMaR project (Stede et al., 2006), which combines the results of various statistical and symbolic analysis modules to compute an informative (extractive) summary of the input text. The MOTS workbench relies on two mechanisms: (1) a generic merging script that converts the PAULA standoff data to a standard inline XML representation, used for effective visualization of the various analysis results; (2) a Java API that allows uniform access to the PAULA data and permits construction of additional layers of analysis. One such layer we are constructing on the basis of "lower-level" input layers is that of local coherence analysis.

[3] `http://www.sfb632.uni-potsdam.de/ANNIS`

5 Local Coherence Analysis

Having described our general approach to automatic text analysis, we now focus our attention on one particular subtask: hypothesizing coherence relations between adjacent spans of text. This is a central aspect of computing text meaning with "deep" approaches, but it is also relevant for robust applications based on "surfacy" methods. For example, Marcu (2000) demonstrated that an algorithm using patterns operating on the surface string can to a certain extent identify the "rhetorical structure" of the text and use this information for the purpose of automatic summarization.

Within the step of coherence analysis, the key role is played by *connectives*: lexical units with a relational meaning that contribute to cohesion and coherence by indicating a connection between adjacent text segments. In the following, Subsection 5.1 first briefly introduces the task of local coherence analysis, and then 5.2 looks in more detail into the treatment of connectives, using a dedicated lexical resource holding information about them for a variety of purposes and applications (5.3). Finally, subsection 5.4 will discuss the embedding of coherence analysis in a multi-level analysis framework.

5.1 "Rhetorical Parsing": The Idea

Among discourse researchers, it is generally taken for granted that *coherence relations*, semantic or pragmatic relationships between (mostly adjacent) text segments, are (besides coreference) a central aspect of text coherence. Similarly, there is agreement that connectives are the primary linguistic means for signalling such relations at the linguistic surface. Views differ, however, on the number and definitions of relations, and also on the formal properties of the structures resulting from assigning coherence relations first to "minimal units" of text and then recursively to larger segments. For different views, see Polanyi (1988), Mann and Thompson (1988), Asher and Lascarides (2003), Wolf and Gibson (2005). For our purposes here, we focus on the role of coherence analysis within the larger enterprise of document processing. Some researchers take the position that coherence relations can be computed all the way from minimal units of analysis up to the document level, i.e., that the relations also hold between paragraphs, sections, and so forth. For certain types of text genre, this is certainly a feasible assumption, and Lüngen et al. (2006) have shown that this approach can be implemented to account for the structure of certain kinds of scientific papers. In general, however, it seems useful to distinguish between phenomena of *local* coherence (viz. semantic or pragmatic relationships between adjacent spans of text) and the *global* structure of a document. The latter is largely determined "top-down" by genre-dependent conventions and schema-like structuring principles; the former arises "bottom-up" when understanding the connections between clauses and sentences. We thus use the term 'local coherence analysis' for the task of identifying such connections, and regard this task as usually being applicable within paragraphs. Occasionally it may very well happen that some coherence relation applies to join neighbouring paragraphs — but in general, we surmise that different types of

relationships hold between larger units of text on the one hand, and between clauses and sentences on the other.

Turning then to the local level, our primary source of information are connectives. These are lexical items belonging to different morphosyntactic classes (conjunctions, adverbials, prepositions) that can either explicitly signal a semantic relationship between text segments (e.g., *nonetheless* quite clearly signals a concessive relationship) or merely invite the reader to construct the type of relationship (e.g., *but* signals some sort of adversarial relation such as contrast, concession, substitution, or correction; *and* is much more general in meaning). Linguists have undertaken many detailled investigations of individual connectives or groups thereof; for German, an invaluable resource to be mentioned here is (Pasch et al., 2003), which defines connectives by means of five features: they are not inflectable, do not assign case to their syntactic environment, denote two-place relations, take as arguments states of affairs *(Sachverhalte)*, which must be expressable as sentences. When, as several authors do, prepositions are added to the class of connectives, the second criterion needs to be weakened.

In automatic text analysis, connectives have been employed, *inter alia*, by Sumita et al. (1992), Corston-Oliver (1998), and Marcu (2000). Marcu's system, as mentioned above, operated on the text surface: He had rules for disambiguating punctuation symbols in order to segment the text into minimal units (sentences or clauses), and then associated connectives with coherence relations to obtain structures according to Rhetorical Structure Theory Mann and Thompson (1988). We illustrate the approach with this short text:[4].

> Because well-formed XML does not permit raw less-than signs and ampersands, if you use a character reference such as `&#60;` or the entity reference `<` to insert the < character, the formatter will output `<` or perhaps `&#60;`.

Supposing that we are able to identify the connectives and punctuation symbols correctly (here in particular: note that *to* is not a spatial preposition; distinguish between commas in enumerations and those finishing clauses), we can identify the "scaffold" of this short text as the following:

```
Because A, if B or C to D, E or F
```

with A to F representing the minimal units of analysis. Next, fairly simple rules will be sufficient to guess the most likely overall bracketing of this string:

```
(Because A, (if ((B or C) to D)), (E or F))
```

And finally, it happens that the connectives *because, if, to* and *or* are quite reliable signals of the coherence relations *Reason, Condition, Purpose* and *Disjunction*, respectively. Combining this information with the bracketing, we obtain the tree structure (in spirit of RST) shown in Figure 1.

Obviously, few texts behave as nicely as the one we just investigated. For one thing, it is known that most coherence relations are *not* explicitly signalled at the text surface. And furthermore, even if a connective is present, we have to reckon with several problems, to which we will attend in the next subsection.

[4]Source: `http://www.cafeconleche.org/books/bible2/chapters/ch17.html`

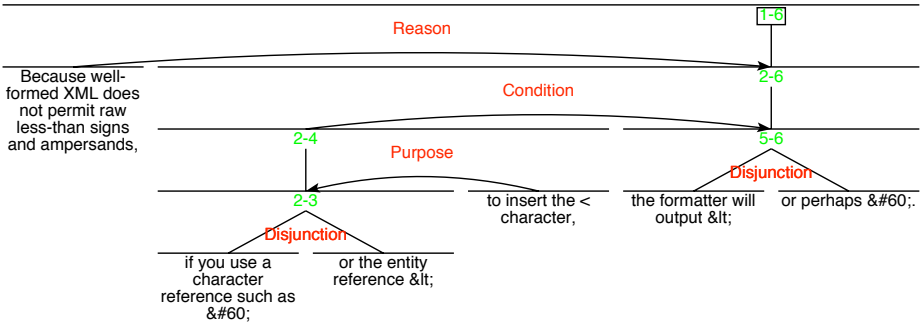


Figure 1: RST-style analysis of sample text

5.2 A Closer Look at Connectives

A closer investigation of the behaviour of connectives in texts reveals a range of complications that can disturb the very parse-friendly situation we encountered in the example above. In particular, we propose twelve phenomena that need to be accounted for; they are associated with the form and syntactic behaviour of connectives, with their basic meaning (in terms of coherence relations) and distinctive pragmatic features, and also with certain discourse-structural implications.

(1) Ambiguity: connective or not. Quite a few German words that can be used as connectives also have other, non-connective readings. For instance, *da* can be a causal subordinator (i.e., connective) but also a locative or temporal anaphor. This type of ambiguity is more widespread than one might think; in Dipper and Stede (2006), we report that 42 out of 135 frequent German connectives also have a non-connective reading, and we point out that many of the problems cannot be handled with off-the-shelf part-of-speech taggers. Hence, local coherence analysis is in need of a disambiguation step.

(2) Ambiguity: connectives can have more than one meaning, i.e., signal more than one coherence relation. For example, *schließlich* can close off a presentational list or enumeration; or it can indicate the ending of a temporal sequence of events; or it can be an argumentative marker conveying that a presented reason is definitive or self-evident. This also relates to the next point (3): Connectives can operate on different types of linguistic objects. A well-known distinction is that between *semantic* and *pragmatic* coherence relations, with the former holding between reported events in the world and the latter between speech acts performed by the interlocutor. In this regard, the temporal reading of *schließlich* conveys a semantic relation, while the argumentative one signals a pragmatic move. Many authors, however, prefer a tripartite distinction along

the lines of Sweetser (1990), who offers *content*, *epistemic* and *speech-act* relations — see, for example, Blühdorn (2005). (4) Some further pragmatic distinctions are usually not modelled as a difference in coherence relation; a well-known case in point is the difference between *because* and *since*, where only the latter has a tendency to mark the following information as hearer-old (not necessarily discourse-old). Also, connectives can convey largely the same information yet differ in terms of stylistic nuances, for instance in degree of formality; thus a concessive relation in English may be signalled in a rather formal way by using a *notwithstanding* construction.

(5) A feature that is somewhat easier to isolate is the (non-)ability of connectives to occur within the scope of focus particles. An example from German is: *Nur weil es regnet, nehme ich das Auto. / ?Nur da es regnet, nehme ich das Auto.* However, a complication is lurking here as well, because this feature borders on (6) the issue of connectives having two parts. Clear cases are *either .. or* and *if .. then*. For the German version *wenn .. dann*, though, a coherence analyzer must account for the possibility of its occurring in reverse order: *Dann nehme ich eben das Auto, wenn Du so bettelst.* Now, looking at highly frequent collocations such as *even though* or *even if*, it is difficult to decide whether we are dealing with a single-word connective and a focus particle, or with a complex connective; one solution is to check in such cases whether the meaning is in fact derived compositionally and in that case to prefer the focus particle analysis. Next, from "regular" two-word connectives it is a small step to (7) the shady area of *phrasal* connectives, which can allow for almost open-ended variation and modification: *aus diesem Grund / aus diesen Gründen / aus all diesen guten Gründen / ...*

Turning to structural questions, one well-known complication is (8) the embedding of discourse units into one another, which is problematic for straightforward tree representations of text structure: *Gestern habe ich, weil ich etwas krank war, keinen Spaziergang gemacht.* Besides embedding, connectives can (9) occasionally link text segments that are non-adjacent — a phenomenon that has been studied intensively by Wolf and Gibson (2005) and also by Webber et al. (2003). An example from Webber et al.: *John loves Barolo. So he ordered three cases of the '97. But he had to cancel the order because then he discovered he was broke.* Here, the *then* is to be understood as linking the discovery event back to the ordering event rather than to the (adjacent) cancelling. Non-adjacency also leads to the issue of crossing dependencies, which is discussed, *inter alia*, by the two teams of authors mentioned above. It correlates with the problem (10) of two connectives occurring in the same clause, as also exemplified in the *Barolo* example *(because then*), which renders the parsing task significantly more complex than in our "ideal" example of the previous subsection. A slightly different problem is to be found in situations where (11) a coherence relations is signalled twice, by two different connectives, where one typically is to be read cataphorically: *Ich nehme deshalb das Auto, weil Du so bettelst.* This is not quite the same as the two-word connectives in (6), and a coherence analyzer will have to be very careful not to hypothesize two separate causal relationships in such examples.

Finally, (12) certain connectives convey information about the discourse structure *beyond* the local relation between two segments. A case in point is the first word of this paragraph, which not only makes a 'List' or 'Enumeration' relation explicit, but also

provides the information that this very list is now coming to an end. A smart coherence analyzer could thus reduce the search space for linking the subsequent text segment — knowing that it will definitely *not* be part of the same 'List' configuration.

5.3 A Declarative Resource: Discourse Marker Lexicon

The catalogue of potential difficulties given in the previous subsection has demonstrated that local coherence analysis based on connectives can be a quite complicated task. Depending on the specific goals of the overall intended application, not all of the problems will always be relevant or critical, but in general it seems advisable to design an approach that is in principle prepared to deal with such issues. The proposal here is to make use of a declarative resource — a lexicon — for assembling all kinds of information about connectives, and then to have a coherence analysis module peruse as much of this information as it possibly can (or needs). The first version of our *Discourse Marker Lexicon* (DIMLEX) was described in (Stede and Umbach, 1998); it was used at the time for relatively simple "rhetorical parsing" as outlined in Section 5.1 and also for a language generation application, where the task is to select for a given coherence relation (as determined by the text planner) a connective that can suitably express that relation in the linguistic context generated so far. To deal with such diverse applications, we followed the basic idea to encode the information in the XML-based DIMLEX in a rather abstract fashion, and to use XSLT scripts to transform it into application-specific versions. During this step, some information from DIMLEX will be ignored, other will be converted to a specific format. For instance, our first rhetorical parser needed the information in Prolog format, while the generator needed Lisp notation; the XSLT scripts produced both of these (and, in addition, HTML versions for visualization purposes) from the "master lexicon".

More recently, Dipper and Stede (2006) worked on the problem of disambiguating (non-)connective uses (complication (1) in the previous subsection), and found that to a large extent, it can be solved by inspecting the local part-of-speech context; we have begun to add corresponding disambiguation rules to DIMLEX. Furthermore, in an ongoing joint project with a group from IDS Mannheim, we are analyzing especially the *causal* connectives in greater detail than we had done before. The results are also being incorporated into the lexicon.

At the moment, a DIMLEX entry holds the following information: (1) ortographic variants of the connective; (2) non-/connective disambiguation rules in the shape of weighted part-of-speech sequences that either favour the connective reading (positive weights) or the non-connective reading (negative weights); (3) connective can be in the scope of a focus particle (yes/no); (4) connective can function as a "correlate" to a different connective (yes/no, and which kinds); (5) syntactic category and features, esp. constraints on positioning within the sentence (represented along the lines of Pasch et al. (2003)) and constraints on the linear order of the *internal* segment (the one containing the connective) and the *external* one; (6) semantic readings: coherence relations that the connective can signal, along with disambiguation information (see below); (7) argument

linking: mapping between internal/external arguments and the 'thematic roles' (example: for *although*, the internal argument is the 'Conceded', the external argument the 'Anyway' participant); (8) semantic or pragmatic features to make fine-grained distinctions between similar connectives.

The information for disambiguating between different coherence relations (6) is, similar to (2), represented as weighted rules. The features used here largely pertain to position, tense and aspect of the clause, mood and modality, or lexical collocations. Weights are derived by corpus analyses and thus reflect the probability that a certain feature co-occurs (or does not co-occur) with a certain relation.

Furthermore, several areas in DIMLEX contain linguistic examples to illustrate the relevant distinctions. The information in DIMLEX is formalized to different degrees: Areas under development consist of natural language descriptions that are gradually being turned into interpretable attribute/value representations when the descriptions become stable. Areas that have reached this stage can be translated, e.g., via XSLT, to a specific application-oriented lexical resource, such as one supporting a local coherence analyzer.

5.4 Local Coherence Analysis in a Multi-Level Approach

Finally, we describe our ongoing re-implementation of the coherence analysis module mentioned earlier, now couched in the multi-level framework, using PAULA representations and the Java API (see Section 4.2). For reasons that should have become clear, we regard automatic coherence analysis as a task that can only be partially solved by current language technology, and accordingly, we view connective-based analysis as one contribution to this effort. Thus our module will generate hypotheses of coherence relations and related spans, solely on the basis of connectives occuring in the text. This information is represented in two PAULA layers and may later be combined with the results of other modules contributing to the task — for instance a module checking for lexical cohesion in order to hypothesize Elaboration relationships, which typically are not signalled by connectives. Modules following in the processing chain may combine the various hypotheses into the most likely overall relational tree structure for the paragraph (or a set of such tree structures, see (Reitter and Stede, 2003)), or they may use the hypotheses directly for some application purpose such as text summarization or question-answering, without relying on a spanning tree.

The first step consists in identifying the connectives: Words listed in DIMLEX are isolated in the text (including a check for complex connectives, i.e., two corresponding words in adjacent clauses), and the disambiguation patterns are checked against the PoS layer. A new layer is created, holding those words that were recognized as connectives. Next, the segmentation module tries to identify the minimal discourse units. It uses three layers: the results of sentence splitting and those of dependency parsing in order to identify clauses functioning as separate units; furthermore, prepositional phrases are isolated as minimal units when their head corresponds to a connective as recorded on the newly created connective layer. The minimal units are in turn represented as a separate layer in PAULA.

Next, the connective layer is extended with information on relations and scopes: Every connective is associated with one or more atrribute-value structures listing possible coherence relations along with probabilities, as well as one or more scope assignments, that is mappings between the relation's thematic roles and lists of minimal unit identifiers. All relations stored with the connective in DIMLEX are recorded as hypotheses, and probabilities added as the result of evaluating the associated disambiguation rules, which largely operate on the syntax layer. For example, for the connective *schließlich* we found that with the main verb of the clause elided, the Reason reading is very unlikely; on the other hand, if the verb is in present tense and the Aktionsart is state, it very likely signals Reason. All rules of this kind are being checked for each connective and a ranking of the associated coherence relations is determined by accumulating the weights.

Finally, for each relation we also hypothesize its scope: The thematic roles are associated with lists of minimal units. Also, in this step we check for the possibilities of correlates: when two connectives appear in the same sentence and can signal the same relation, and (according to the DIMLEX entry) one could be a correlate of the other, it is marked as such. Scope determination is usually straightforward for coordinating and subordinating conjunctions. For adverbials, we typically hypothesize different solutions and rank them according to size: The most narrow interpretation is taken as most likely. In this step, we consider the analysis layer of logical document structure in order to disprefer segments that would stretch across paragraphs or other kinds of boundaries. Similarly, a layer with the results of "text tiling" (breakdown of the text in terms of thematic units, in the tradition of Hearst (1994)) can be used for this purpose, as well as as an "attribution" layer that identifies those modal contexts that attribute a span of text to a particular source (as in indirect speech).

For illustration, we consider the first half of a text from the *Potsdam Commentary Corpus* (Stede, 2004), which argues against preserving a disputed building in Berlin, the so-called "Steglitzer Kreisel" (see Figure 2). Connective identification would isolate the words that are set in italics, resulting in the "scaffold" for the text:

```
(1). Selbst wenn (2). (3). Aber (4). (5). Nicht nur (6a),
sondern (6b). Zwar (7). Aber (8).
```

Selbst wenn is analyzed as a connective with focus particle; since no *dann* is present in the subsequent clause, the (infrequent) linear order *"A. Selbst wenn B."* can be recognized as such. The text contains two examples of complex connectives: *nicht nur . . . sondern* in (6), and *zwar . . . aber* in (7/8). The former suggests only one scope assignment; as for the latter, one segment is trivially the *zwar*-sentence, while the other probably ends with (8), but it might also extend beyond that sentence. Thus we need to represent alternative scope assignments. The same holds for the *aber* in (4): The left segment obviously includes (3) but can extend to (2) and (1). Similarly, the right segment can consist of merely (4) or more — the connective analysis procedure cannot make a decision here but has to represent the (weighted) alternatives. As for the relations, we (depending of course on the inventory we use) encounter ambiguities for *aber* and *zwar aber*; in terms of Mann and Thompson (1988), the former can signal Contrast, Antithesis or Concession; the latter only Antithesis and Concession (*zwar* clearly marks a satellite in RST terms, and thus the

(1) Alles spricht gegen den Steglitzer Kreisel. (2) *Selbst wenn* man vergisst, dass der olle Schuhkarton in bester Lage einst ein privates Prestigeobjekt war, das der öffentlichen Hand für teures Geld aufgenötigt wurde. (3) Ein Symbol der West-Berliner Filzwirtschaft in den späten sechziger Jahren. (4) *Aber* lassen wir das ruhig beiseite. (5) Der Kreisel ist Asbest verseucht. (6) *Nicht nur* hier und da, *sondern* durch und durch. (7) *Zwar* könnte man, wie beim Palast der Republik, den Bau bis aufs wackelige Stahlskelett entkleiden und neu aufbauen. (8) *Aber* das würde mindestens 84 Millionen Euro, vielleicht auch das Doppelte kosten. (…)

Figure 2: Excerpt from sample text (Source: *Tagesspiegel*). Sentence numbers added for reference.

multinuclear Contrast does not apply). In these cases, the disambiguation procedure will not be able to distinguish between these (very similar) relations.

6 Looking back and forth – Conclusion

Having described the multi-level approach as a text-technological view on the overall problem of text analysis, we can now compare it to the 'historical' perspectives characterized in Sections 2 and 3. First, notice that the MLA approach is not in conflict with knowledge-intensive processes: ontologies, inference engines, and/or rich lexical resources can of course contribute to modules analyzing phenomena like coreference between definite NPs, coherence relations, etc. There is, however, a sharp contrast to the "knowledge-only" approaches in the early Conceptual Dependency tradition: MLA is based on the assumptions that text analysis should not be an all-or-nothing step, and that it should not *necessarily* rely on pre-coded knowledge of the domain or the world-at-large. The radical modularity and emphasis on re-combinability of modules also differentiates MLA from LILOG-style pipeline architectures, where interfaces between subsequent modules were fine-tuned in order to achieve a smooth interaction of, for instance, sentence syntax and subsequent semantic interpretation, so that certain interesting linguistic phenomena could be handled. MLA instead emphasizes the independence of modules, which entails that more difficult work needs to be done by the "high-level" modules when combining the, possibly partial or even conflicting, results of "lower-level" modules. An approach that, like LILOG, focuses on the linguistic (sentence-based) analysis but shares many of MLA's goals is the 'Heart of Gold' architecture (Schäfer, 2007), which integrates tagging, chunking, parsing and other modules, and represents the — possibly underspecified — results using minimal-recursion semantics (Copestake et al., 2005).

However, MLA (being a text-technological conception) aims to cover text documents as a whole and thus to also account for (logical and content-based) document structure and for phenomena of discourse structure. As a case in point, we took the task of

local coherence analysis, which on the one hand builds upon other analysis levels (part-of-speech tagging, syntactic analysis, logical document structure, text tiling) and then contributes a new level consisting of two technical layers: One dividing the text into a sequence of minimal units, the other identifying the connectives and associating them with (possibly sets of) coherence relations and scope assignments. Thus we do not aim at constructing a full "discourse tree" in this step. For this more ambitious task, other modules could contribute their share of information, such as a lexical-cohesion module delivering hypotheses on Elaboration relations. Then, one can join the accumulated information into the most likely overall tree for a paragraph, or alternatively peruse the individual pieces of information in isolation.

The main advantage of MLA as described here is that of "radical modularity": A framework such as that of PAULA makes it very simple to add a new module or to replace an existing one with a better one fulfilling the same function. Or, for that matter, to employ several alternative modules with the same function (e.g., part-of-speech taggers) in order to evaluate them in the context of a specific application, or to implement voting schemes for a particular level of analysis. At the same time, this great flexibility comes with a price tag: Processing many levels of standoff-XML annotations is computationally expensive. For specific applications, effective programming-language-specific APIs thus play an important role in reducing the need for traversing the graph structures with generic XML processing tools. As pointed out above, we have developed a Java API for the PAULA framework; further APIs for script languages are in preparation.

References

Asher, N. and Lascarides, A. (2003). *Logics of Conversation*. Cambridge University Press, Cambridge.

Blühdorn, H. (2005). Zur Semantik kausaler Satzverbindungen: Integration, Fokussierung, Definitheit und modale Umgebung. *Studi Linguistici e Filologici Online*, 3(2):311–338.

Chiarcos, C., Dipper, S., Götze, M., Ritz, J., and Stede, M. (2008). A flexible framework for integrating annotations from different tools and tagsets. In *Proc. of the First International Conference on Global Interoperability for Language Resources*, Hongkong.

Copestake, A., Flickinger, D., Sag, I., and Pollard, C. (2005). Minimal recursion semantics: An introduction. *Journal of Research on Language and Computation*, 3(3):281–332.

Corston-Oliver, S. (1998). *Computing of Representations of the Structure of Written Discourse*. PhD thesis, University of California at Santa Barbara.

Cunningham, H., Maynard, D., Bontcheva, K., and Tablan, V. (2002). GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*.

DeJong, G. (1982). An overview of the FRUMP system. In Lehnert, W. and Ringle, M., editors, *Strategies for Natural Language Processing*, pages 149–176. Lawrence Erlbaum Associates, Hillsdale/NJ.

Dipper, S. (2005). XML-based stand-off representation and exploitation of multi-level linguistic annotation. In Eckstein, R. and Tolksdorf, R., editors, *Proceedings of Berliner XML Tage*, pages 39–50.

Dipper, S. and Stede, M. (2006). Disambiguating potential connectives. In Butt, M., editor, *Proceedings of KONVENS '06*, pages 167–173, Konstanz.

Endriss, U., Küssner, U., and Stede, M. (1998). Repräsentation zeitlicher Ausdrücke: Die Temporal Expression Language. Verbmobil Memo 133, Technical University Berlin, Department of Computer Science.

Goecke, D., Naber, D., and Witt, A. (2003). Query von Multiebenen-annotierten XML-dokumenten mit Prolog. In Seewald-Heeg, U., editor, *Sprachtechnologie für die multilinguale Kommunikation*. Gardez! Verlag, Sankt Augustin.

Götz, T. and Suhre, O. (2004). Design and implementation of the UIMA common analysis system. *IBM Systems Journal*, 43(3).

Grishman, R. and Sundheim, B. (1996). Message understanding conference - 6: A brief history. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING)*, pages 466–471, Kopenhagen.

Hearst, M. A. (1994). Multi-paragraph segmentation of expository text. In *Proceedings of the 32nd Meeting of the Association for Computational Linguistics*, Las Cruces/NM.

Herzog, O. and Rollinger, C.-R., editors (1991). *Text Understanding in LILOG: Integrating Computational Linguistics and Artificial Intelligence*. Springer, Berlin/Heidelberg.

Hirst, G. and Ryan, M. (1992). Mixed-depth representations for natural language text. In Jacobs, P., editor, *Text-Based Intelligent Systems*, pages 59–82. Lawrence Erlbaum Associates, Hillsdale/NJ.

Hovy, E., Hermjakob, U., Lin, C.-Y., and Ravichandran, D. (2002). Using knowledge to facilitate pinpointing of factoid answers. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING)*, Taipei.

Lobin, H. (2000). *Informationsmodellierung in XML und SGML*. Springer, Berlin/Heidelberg.

Luft, A. (2007). Automatische erkennung und annotierung der temporalen struktur in texten. Diplomarbeit, Hochschule Mittweida, Fachbereich Mathematik/Physik/Informatik.

Lüngen, H., Lobin, H., Bärenfänger, M., Hilbert, M., and Puskas, C. (2006). Text parsing of a complex genre. In Martens, B. and Dobreva, M., editors, *Proc. of the Conference on Electronic Publishing (ELPUB 2006)*, Bansko, Bulgaria.

Mann, W. and Thompson, S. (1988). Rhetorical structure theory: Towards a functional theory of text organization. *TEXT*, 8:243–281.

Marcu, D. (2000). *The theory and practice of discourse parsing and summarization*. MIT Press, Cambridge/MA.

Pasch, R., Braußе, U., Breindl, E., and Waßner, U. H. (2003). *Handbuch der deutschen Konnektoren*. Walter de Gruyter, Berlin/New York.

Polanyi, L. (1988). A formal model of the structure of discourse. *Journal of Pragmatics*, 12:601–638.

Reiter, E. (2007). Last word: The shrinking horizons of computational linguistics. *Computational Linguistics*, 33(2):283–287.

Reitter, D. and Stede, M. (2003). Step by step: underspecified markup in incremental rhetorical analysis. In *Proceedings of the 4th International Workshop on Linguistically Interpreted Corpora (LINC)*, Budapest.

Schäfer, U. (2007). *Integrating Deep and Shallow Natural Language Processing Components - Representations and Hybrid Architectures*. PhD thesis, Saarland University. Vol. 22 of the Saarbrücken Dissertation Series in Computational Linguistics and Language Technology.

Schank, R. C. and Riesbeck, C. K. (1981). *Inside Computer Understanding: Five Programs Plus Miniatures*. Lawrence Erlbaum Associates, Hillsdale/NJ.

Stede, M. (1992). The search for robustness in natural language understanding. *Artificial Intelligence Review*, 6(4):383–414.

Stede, M. (2004). The Potsdam commentary corpus. In *Proceedings of the ACL Workshop on Discourse Annotation*, pages 96–102, Barcelona.

Stede, M., Bieler, H., Dipper, S., and Suryiawongkul, A. (2006). SUMMaR: Combining linguistics and statistics for text summarization. In *Proceedings of the European Conference on Artificial Intelligence*, Riva del Garda.

Stede, M. and Umbach, C. (1998). DiMLex: A lexicon of discourse markers for text generation and understanding. In *Proceedings of the 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics (COLING/ACL'98)*, pages 1238–1242, Montreal, Canada.

Sumita, K., Ono, K., Chino, T., Ukita, T., and Amano, S. (1992). A discourse structure analyzer for Japanese text. In *Proceedings of the International Conference on Fifth Generation Computer Systems*, pages 1133–1140.

Sweetser, E. (1990). *From etymology to pragmatics*. Cambridge University Press, Cambridge.

Webber, B., Stone, M., Joshi, A., and Knott, A. (2003). Anaphora and discourse structure. *Computational Linguistics*, 29(4):545–587.

Wiebe, J. (1994). Tracking point of view in narrative. *Computational Linguistics*, 20(2):233–287.

Wolf, F. and Gibson, E. (2005). Representing discourse coherence: a corpus-based study. *Computational Linguistics*, 31(2):249–287.

Caroline Sporleder

Lexical Models to Identify Unmarked Discourse Relations: Does WordNet help?

Abstract

In this paper, we address the task of automatically determining which discourse relation holds between two text spans. We focus on relations that are not explicitly signalled by a discourse marker like *but*. While lexical models have been found useful for the task, they are also prone to data sparseness problems, which is a big drawback given the scarcity of discourse annotated data. We therefore investigate whether the use of lexical-semantic resources, such as WordNet, can be exploited to back-off to a more general representation of lexical information in cases were data are sparse. We compare such a semantic back-off strategy to morphological generalisations over word forms, such as stemming and lemmatising.

1 Introduction

To be able to interpret a text it is important to know how its sentences and clauses relate to each other. For example, whether the events referred to stand in a causal relation or whether one text segment provides an elaboration or a summary of another. This type of information is also crucial for many natural language processing (NLP) tasks. Question answering, for instance, frequently involves recognising cause and effect, e.g., to answer questions like "*Why did Romano Prodi resign?*" or "*What is the effect of Benzodiazepines in elderly people?*". Likewise, text summarisation systems need to know which pieces of information in a text are essential and which ones merely elaborate.

While there has been a considerable research effort dedicated to the automatic identification of discourse relations between text segments, the problem is still far from being solved, with state-of-the-art systems typically obtaining F-Scores between 40% and 70%, depending on the exact task and the number of discourse relations considered (see, e.g., Marcu (2000); Soricut and Marcu (2003); Le Thanh et al. (2004); Pardo et al. (2004); Baldridge and Lascarides (2005); Baldridge et al. (2007)). Moreover, most approaches heavily rely on surface cues, especially the presence of overt discourse markers such as *because* or *but*. Few systems have been dedicated to determine relations *in the absence* of such markers.[1] However, it has been estimated that only around half of all relations are explicitly signalled by a discourse connective (Redeker (1990); Eugenio et al. (1997); Marcu (2000)). Connectives are also often ambiguous, either between discourse usage and non-discourse usage (e.g., *for* as a synonym of *because* vs. *for* as a preposition) or be-

[1] A notable exception are Marcu and Echihabi (2002).

tween two or more discourse relations (e.g., *since* can signal a temporal or an explanation relation). Effectively, one can distinguish three, progressively more difficult, cases:

1. a relation is signalled by an **unambiguous marker**
2. a relation is signalled by an **ambiguous marker**
3. a relation is **not explicitly signalled** by any marker

Relations falling in the first set can be trivially identified provided one has a list mapping unambiguous markers to the relations they signal.[2] For the second case, discourse markers have to be disambiguated. For the third case, relations need to be identified based on other cues, such as the lexical semantics of the words in the sentences. The performance on the third task is likely to be much lower than the F-Scores of 40%-70% reported above for systems that address all types of relations. Identifying discourse relations which are not signalled by explicit discourse markers is thus one of the main bottlenecks for the automatic determination of discourse structure.

In this paper, we focus specifically on distinguishing *unmarked* discourse relations, which we define as covering both, relations which are signalled by ambiguous markers (case two above) and relations which are not signalled by any discourse markers (case three). The reason for not distinguishing between these two cases is that it is sometimes difficult to tell whether a relation is ambiguously signalled or not at all; some discourse connectives, such as *and*, are so ambiguous with respect to the relations they can signal that they supply hardly any discourse information at all.

While we do not aim at *solving* the task of recognising unmarked relations in this paper, we intend to shed some light on *lexical cues* that can or cannot help to identify such relations. Intuitively, lexical information provides useful cues for this task, as the correct discourse relation can often be guessed on the basis of the lexical semantics of the words involved. For instance, the two spans (marked by square brackets) in example (1) are related by EXPLANATION and a human may already be able to infer this from the words *late* and *missed the bus* alone. Likewise, the CONTRAST relation in example (2) (taken from Marcu and Echihabi (2002)) can be guessed from the presence of the two words *good* and *fail* which indicate a contrast. Similarly, in example (3) the SUMMARY relation might be inferable from the occurrence of *expensive* and *$7,000* in the left and right spans, respectively.

(1) [Peter was late this morning,] [he had missed the bus.]
(EXPLANATION)

(2) [Paul is *good* in maths and sciences.] [Peter *fails* almost every class he takes.]
(CONTRAST)

[2] The set of unambiguous markers depends to some extent on the discourse theory that is used. For example *in other words* can signal either RESTATEMENT or SUMMARY in *Rhetorical Structure Theory* (RST, Mann and Thompson (1987)), whereas it unambiguously signals SUMMARY in *Segmented Discourse Representation Theory* (SDRT, Asher and Lascarides (2003)) because the latter theory does not distinguish these two relations.

(3) ["It may be very *expensive*," the spokesman warned.] ["The price cannot be less than *$7,000*."]
(SUMMARY)

Empirical evidence for the importance of lexical information for identifying discourse relations has also been provided by a number of previous studies. Virtually all data-driven approaches to discourse parsing employ some lexical information to determine discourse relations. Polanyi et al. (2004), for example, make use of information about lexeme repetition and synonym, antonym, and hypernym relationships between lexemes, in addition to other cues (syntactic and structural information) to determine discourse relations. Forbes et al. (2001) rely heavily on lexicalised tree fragments to derive discourse structure. Likewise Soricut and Marcu (2003) propose a lexicalised discourse parser. Le Thanh et al. (2004) exploit lexical and syntactic cues to build their discourse trees. The system by Pardo et al. (2004) is completely based on surface cues and does not require syntactic information, relying solely on discourse markers and cue words. Similarly, Marcu and Echihabi (2002) determine the discourse relations holding between two spans solely on the basis of the words occurring in the spans. Finally, Sporleder and Lascarides (2005) found that lexical cues were among the best performing features in their multi-feature system for determining discourse relations.

While lexical cues can contribute in identifying the correct discourse relation in some examples, lexical cues also tend to be prone to data sparseness. The reliable learning of a mapping from lexical properties to discourse relations typically requires a very large amount of annotated data for training. Unfortunately, the training sets available are normally fairly small as annotated data is expensive to create. Marcu and Echihabi (2002) proposed to address the lack of training data by automatically creating labelled data from unannotated corpora. For this, they extracted unambiguously marked examples from a corpus, labelled them with the relation signalled by the marker, then removed the marker and trained a lexical model to recognise discourse relations in the absence of any marker. However, their approach was found not to generalise very well to naturally unmarked instances (Murray et al., 2006; Blair-Goldensohn et al., 2007; Sporleder and Lascarides, 2008).

An alternative to increasing the annotated data by automatic example labelling is to look for a representation of lexical information that is less prone to sparse data problems. For NLP tasks such as prepositional phrase attachment (Clark and Weir, 2000) or compound noun analysis (Nastase et al., 2006), it has been suggested to replace individual lexical items by more general classes, such as hypernyms taken from WordNet (Miller et al., 1990), in order to overcome data sparseness. In this paper, we investigate whether class-based information is also useful for identifying discourse relations and how this strategy compares to other methods of generalising over the actual word forms, such as lemmatising or stemming.

2 Experimental Set-up

To determine which of the generalisation strategies performs best, we first created a data set of pairs of text spans which are linked by unmarked discourse relations. We then created a number of two-feature classifiers, in which one feature encoded information about the left span at a given level of generalisation and the second feature encoded the same type of information for the right span. For example, the first feature might encode the stems in the left span and the second the stems in the right span. To determine the utility of each feature type, we ran a 10-fold cross-validation experiment for each of the classifiers in isolation. We also assessed the data sparseness that resulted from a particular encoding of the spans.

The next section describes the data creation in more detail. Section 2.2 outlines the machine learning framework we employed and 2.3 lists the individual features we tested.

2.1 Data

For our experiments, we looked at five relations from *Segmented Discourse Representation Theory* (SDRT, Asher and Lascarides (2003)): CONTRAST, EXPLANATION, RESULT, SUMMARY, and CONTINUATION. SDRT relations tend to be more coarsely-grained than those used by Rhetorical Structure Theory (RST, (Mann and Thompson, 1987)) and are therefore more amenable to automatic analysis. Examples of the five relations are given below (examples 4 to 8). For a detailed definition of each of the relations see Asher and Lascarides (2003).

(4) [The executive said any buy-out would be led by the current board, whose chairman is Maurice Saatchi and whose strategic guiding force is believed to be Charles Saatchi.]
[Mr. Spielvogel isn't part of the board, nor are any of the other heads of Saatchi's big U.S.-based ad agencies.]
(CONTRAST)

(5) [The five astronauts returned to Earth about three hours early because high winds had been predicted at the landing site.]
[Fog shrouded the base before touchdown.]
(CONTINUATION)

(6) [The venture's importance for Thomson is great.]
[Thomson feels the future of its defense business depends on building cooperation with other Europeans.]
(EXPLANATION)

(7) [A broker may have to approach as many as 20 underwriters who insure the endeavors on behalf of the syndicates.]
[It could take six months for a claim to be paid.]
(RESULT)

(8) ["It will be very expensive," the spokesman warned.]
["The price cannot be less than $7,000."]
(SUMMARY)

To create the data set, we collected examples from the RST Discourse Treebank (RST-DT, Carlson et al. (2002)) and manually mapped them to SDRT relations (see Sporleder and Lascarides (2008) for details). We only extracted examples in which the relation is not signalled by an unambiguous discourse marker and in which the relation holds between the clauses of a sentence or between adjacent sentences; we did not collect relations holding between multi-sentence text spans.[3] Overall, our final data set contained 1,051 examples, with roughly equal proportions of all relations with the exception of SUMMARY for which we found only 44 examples in the RST-DT (see Table 1). The inter-annotator agreement for identifying the relations was 72% (kappa .592, Carletta (1996)). This is effectively an upper bound on the performance we can expect from automatic relation classifiers. The fact that the agreement is noticeably below 100% also shows that the task of classifying discourse relations in the absence of unambiguous markers is difficult even for humans.

Relation	number of examples
CONTRAST	213
EXPLANATION	268
CONTINUATION	260
RESULT	266
SUMMARY	44

Table 1: Examples per relation in the data set

2.2 Machine Learning Framework

We chose BoosTexter (Schapire and Singer, 2000) as our machine learner. BoosTexter was originally developed for text categorisation. It combines a boosting algorithm with simple decision rules and allows a variety of feature types, such as nominal, numerical or text-valued features. Text-valued features can, for instance, encode sequences of words or parts-of-speech. BoosTexter applies statistical models to automatically identify informative n-grams when forming classification hypotheses for these features (i.e., it tries to detect n-grams in the sequence which are good predictors for a given class label). BoosTexter's effective modelling of n-gram features makes it particularly suitable for our

[3] One reason for excluding the latter is that relations between larger text spans are distributed differently than relations between sentences or clauses, e.g., RST relations like ELABORATION, JOINT, and BACKGROUND are more frequent between larger units than between sentences and clauses whereas relations like CONTRAST, RESULT, and EXPLANATION are more frequent between smaller units. Consequently relations between larger units are often treated by different means than inter- or intra-sentential relations (see e.g. Marcu (2000)).

task as we can directly encode the words, stems, hypernyms etc. of the two text spans involved in a relation as text-valued features. In addition to supporting n-gram features, BoosTexter also allows the use of *sparse n-grams*, i.e. n-grams with variable slots. For instance, the sparse n-gram *Dow Jones * sank* would match among others the 4-grams *Dow Jones Industrials sank* and *Dow Jones index sank*. We experimented with both, normal and sparse n-grams up to $n = 3$ and $n = 4$. The next section lists the features in detail.

2.3 Lexical Features

We implemented 10 lexical feature pairs (with one feature for the left and the other for the right span), encoding tokens (with and without punctuation and stop words), stems, lemmas, content word lemmas, word sense disambiguated lemmas, and hypernyms. To extract this information, we employed a number of pre-processing tools: Tokenisation, lemmatisation, and part-of-speech tagging were done by the tools supplied with the RASP parser (Briscoe et al., 2006).[4] Stemming was performed by applying the Porter stemmer (Porter, 1980). For the hypernym back-off we needed to word sense disambiguate the data. This was done by employing the SenseRelate disambiguation package (Pedersen et al., 2005). In this approach a target word is disambiguated by computing the semantic relatedness between each of its possible senses and all possible senses of the neighbouring words, and then choosing the sense that gives rise to the highest relatedness score. Semantic relatedness between two senses is computed by looking at their gloss overlap in WordNet 2.0 (Fellbaum, 1998). Below, we discuss the features in more detail, using the span pair in example (9) for illustration, where *(LS)* and *(RS)* indicated the left and right span, respectively.

(9) (LS) A broker may have to approach as many as 20 underwriters who insure the endeavors on behalf of the syndicates.
(RS) It could take six months for a claim to be paid.
(RESULT)

Words: encodes the spans as they occur in the text after tokenisation and normalising capitalisation:

(10) (LS) a broker may have to approach as many as 20 underwriters who insure the endeavors on behalf of the syndicates .
(RS) it could take six months for a claim to be paid .

We also encoded variants of this feature pair in which punctuation characters and/or stop words were removed.

Lemmas: encodes the original strings with all words lemmatised:

[4] Note that we did not employ full parsing or indeed any syntactic information, such as chunking.

(11) (LS) a broker may have to approach as many as 20 underwriter who insure the endeavor on behalf of the syndicate .
(RS) it can take six month for a claim to be pay .

Stems: encodes the original strings with all words stemmed:

(12) (LS) a broker mai have to approach as mani as 20 underwrit who insur the endeavor on behalf of the syndic .
(RS) it could take six month for a claim to be paid .

Content word lemmas: encodes only content word lemmas in the two spans. Named entities and numbers are replaced by placeholders (NE and NUM, respectively). We identified named entities and numbers from the part-of-speech tagged spans.

(13) (LS) broker approach NUM underwriter insure endeavor syndicate
(RS) take NUM month claim pay

Word sense disambiguated lemmas: encodes all lemmas in the original spans but lemmas are disambiguated where possible (i.e., if the lemma can be found in WordNet):

(14) (LS) a broker#n#1 may#v have#v#13 to approach#v#5 as many as 20 underwriter#n#1 who insure#v#1 the endeavor#n#1 on behalf#n#1 of the syndicate#n#2
(RS) it could#v take#v#10 six month#n#2 for a claim#n#1 to be#v#1 pay#v#8

Hypernym back-off for all word sense disambiguated lemmas: all word sense disambiguated lemmas are replaced by their direct hypernyms in WordNet (example (15)).[5] We also implemented two variants in which we back-off to hypernyms that are two and three levels higher up the hierarchy (see example (16) for a three level back-off).

(15) (LS) a businessperson#n#1 may#v have#v#13 to address#v#9 as many as 20 agent#n#4 who verify#v#1 the undertaking#n#1 on stead#n#1 of the association#n#1
(RS) it could#v decide#v#1 six time_unit#n#1 for a assertion#n#1 to be#v#1 be#v#1

(16) (LS) a person#n#1 may#v have#v#13 to travel#v#1 as many as 20 capitalist#n#2 who confirm#v#1 the activity#n#1 on duty#n#1 of the social_group#n#1
(RS) it could#v decide#v#1 six abstraction#n#6 for a statement#n#1 to be#v#1 be#v#1

[5] The repeated occurrence *be#v#1* at the end of the second span in example (15) can be explained as follows. The first occurrence comes from *be* in *to be paid* for which there are no hypernyms for the assigned sense *be#v#1*. The second occurrence of *be#v#1* comes from the word *pay* which is wrongly disambiguated and assigned the sense used in *it pays to go through trouble*. The direct hypernym of this sense in WordNet 2.0 is also *be#v#1*. Hence the repeated occurrence of this sense at the end of the second span.

Hypernym back-off for infrequent lemmas: for this feature, lemmas are backed-off to their hypernyms if they occur only once in the data set (*hapax legomena*).[6] For example, the lemmas *endeavour* and *syndicate* are replaced by their hypernyms *undertaking* and *association*, respectively.

(17) (LS) a broker#n#1 may#v have#v#13 to approach#v#5 as many as 20 underwriter#n#1 who insure#v#1 the **undertaking#n#1** on behalf#n#1 of the **association#n#1**
(RS) it could#v take#v#10 six month#n#2 for a claim#n#1 to be#v#1 pay#v#8

Placeholder back-off for infrequent lemmas: this feature is a variant of the previous one, hapaxes are replaced by a placeholder (*INFR*) rather than backed-off to the next hypernym level.

(18) (LS) a broker#n#1 may#v have#v#13 to approach#v#5 as many as 20 underwriter#n#1 who insure#v#1 the **INFR** on behalf#n#1 of the **INFR**
(RS) it could#v take#v#10 six month#n#2 for a claim#n#1 to be#v#1 pay#v#8

3 Data sparseness and classification accuracy for different lexical representations

Each of the features described in the previous section is effectively a different representation of the lexical items in the two text strings involved in a discourse relation. To determine the utility of the different representations, we determined their effect on (i) data sparseness and (ii) the accuracy of the relation classifier.

3.1 Data Sparseness

We estimated the data sparseness by computing the type-to-token ratio for different representations of lexical items (words, lemmas, stems, hypernyms, etc.) and the number of hapax legomena, both in absolute terms and in relation to the number of tokens. Table 2 shows the results.

It can be seen that the highest type-to-token ratio is achieved for word strings without punctuation and stop words. The lowest number of hapaxes is predictably achieved for *hapax back-off to placeholder* which eliminates all hapaxes. Note that *hapax back-off to hypernyms* does not eliminate all hapaxes, as we only back-off one level and the hypernym may itself only occur once in the data. Encoding only content lemmas leads to the second lowest number of singular items. Word-sense disambiguation, predictably, leads to an increase in hapaxes, which is then reduced by general hypernym back-off.

3.2 Classification Accuracy

For each lexical representation we trained a two-feature classifier, where the two features corresponded to the right and left spans of the instances in the data set. We then ran four

[6] We also experimented with other frequency thresholds, without much effect on the results.

	type-token ratio	num. of hapaxes	hapax-token ratio
words	15.90% (6241/39240)	3185	8.12%
words no punct.	19.34% (6304/32591)	3185	10.06%
words no punct. no stop	33.33% (6046/18139)	3185	17.92%
stems	13.18% (4994/37888)	2453	6.47%
lemmas	14.68% (5562/37888)	2883	7.61%
content lemmas	18.19% (3001/16500)	1447	8.77%
wsd lemmas	21.80% (7122/32672)	4039	12.36%
hypernyms all, level 1	16.85% (5506/32672)	2967	9.08%
hypernyms all, level 2	14.55% (4755/32672)	2608	7.98%
hypernyms all, level 3	13.23% (4321/32672)	2423	7.42%
hapax back-off hypernyms	19.35% (6323/32672)	3044	9.32%
hapax back-off placeholder	9.44% (3084/32672)	0	0.00%

Table 2: Data sparseness for different lexical features

10-fold cross-validation experiments for each of the 12 two-feature classifiers, using four parameter settings, i.e., n-grams and sparse n-grams up to n=3 and n=4. The average classification accuracies for each run are shown in Table 3.

While the results are all relatively close together and many of the differences are not statistically significant, some trends can be observed. With respect to the different feature types it can be seen that representing only content word lemmas generally leads to the worst results with classification accuracies between 29.29% and 30.68%. Since the classifiers that are based on an encoding that represents *all* lemmas in the spans seem to perform best (with classification accuracies between 43.38% and 45.71%), it can be concluded that non-content word lemmas (e.g. function words) are quite important for the classification task. This conclusion is further corroborated by the fact that the word-based classifier which excludes stop words performs around 10% lower than the one that includes this information. Lemmatising and stemming tend to lead to a higher performance than encoding the words in the spans directly. Word-sense disambiguation leads to a drop in accuracy compared to using the non-disambiguated lemmas but this decrease is quite small for n-grams. The lower accuracies can probably be attributed to the increased data sparseness and also the introduction of noise due to wrongly disambiguated lemmas. Indiscriminant back-off to the next hypernym level leads to a further drop in performance. Only backing-off hapaxes to their hypernyms seems to be a better strategy, though the classifiers that use these features still performed worse than those that employ the disambiguated lemmas without any back-off. Hypernym back-off

	avg. accuracy (%)			
	n-grams		sparse n-grams	
	$n \leq 3$	$n \leq 4$	$n \leq 3$	$n \leq 4$
words	41.87	41.47	43.06	44.07
words, no punct.	43.63	43.63	42.42	42.69
words, no punct. no stop	31.64	31.64	32.18	31.84
stems	43.16	43.75	43.40	43.84
lemmas	43.38	43.77	45.71	45.00
content lemmas	29.29	29.29	30.68	29.51
wsd lemmas	43.15	43.15	41.85	41.32
hypernyms all, level 1	40.35	40.29	40.59	40.01
hypernyms all, level 2	41.39	39.77	39.48	41.39
hypernyms all, level 3	38.33	39.48	38.38	39.40
hapax back-off hypernyms	42.83	41.52	39.99	42.57
hapax back-off placeholder	40.39	40.31	40.49	40.92

Table 3: Classification Accuracies, averaged over ten 10-fold cross-validation runs

tends to perform better than back-off to a simple placeholder. With respect to n-grams versus sparse n-grams, it seems that the latter generally lead to a higher accuracy but this is not true for all features.

On the whole, our results suggest that alleviating data-sparseness by morphological processing, such as stemming or lemmatising, is a more successful strategy than using semantic generalisation strategies, e.g., backing-off to hypernyms. One reason for this is probably that word sense disambiguation is by no means a solved NLP task and state-of-the art disambiguation systems still have a relatively high error rate.[7] Word sense disambiguation thus inevitably introduces noise, and this may outweigh any gains that could potentially be made by semantic back-off strategies. A second reason for the relatively low performance of the WordNet-based features may be that we used a relatively crude back-off strategy. Ideally one would want to automatically determine the right back-off level, i.e., backing-off to a concept that is general enough to reduce sparseness but specific enough to allow the classifier to discriminate between different discourse relations. Sophisticated semantic back-off strategies exist for a number of

[7] The exact proportion of errors depends on several factors, for example on how finely-grained the sense inventory is. One way to verify whether the relatively bad performance of hypernym back-off is indeed due to word sense disambiguation errors would be to re-run the experiments on data with manually disambiguated senses. Unfortunately, manual word sense disambiguation is a very time-consuming task and disambiguating the complete data set was beyond the scope of this paper. However, we manually checked a small sample of the automatically disambiguated data and found a significant proportion of errors (30-40%).

NLP tasks, such as parse disambiguation, (Clark and Weir, 2000, 2002; Li and Abe, 1998; Resnik, 1998). However, these require labelled training data and are therefore difficult to transfer to the task of determining discourse relations for which the amount of labelled training data is very small.

4 Conclusion and Outlook

In this paper we have presented an initial study on the benefit of different lexical representations for the task of classifying unmarked discourse relations. Since lexical models suffer from sparse data we investigated different methods of generalising over the actual word forms in the spans and backing-off to less sparse lexical items. We looked in particular at semantic back-off to hypernyms. Our results suggest that semantic generalisations are considerably less effective than morphological ones, such as lemmatising or stemming. Lemmatisation was found to be the best strategy. We also found that non-content word lemmas play a fairly important role in the classification task and should not be disregarded. The relatively low performance of semantic back-off models is probably largely due to errors in the word-sense disambiguation and possibly also to the difficulty of finding a suitable back-off level automatically.

While the current study focused only on lexical features, future work on the classification of discourse relations in unmarked examples should also take other sources of information into account. The main challenge for this task is to find a good representation of the *meaning* (or the most important aspects thereof) of the two spans involved in a relation. This representation should be general enough so that it minimises data sparseness and specific enough that a machine learning system can learn to discriminate between different relations. The task thus bears similarities to other complex semantic task such as recognising textual entailment (RTE) or finding paraphrases. Though, because the latter tasks aim at estimating semantic *similarity* at some level, some mileage can be gained by relatively simple methods such as word overlap. Most discourse relations, however, cannot be modelled by such simple statistical methods. The most successful RTE systems currently exploit a whole number of external resources, e.g., WordNet, logical inference, anaphora resolution, and large corpora of entailment examples (Hickl and Bensley, 2007; Giampiccolo et al., 2007). It is likely that such a multi-resource strategy is also necessary to successfully distinguish unmarked discourse relations.

Acknowledgements

This work was funded by the German Research Foundation DFG (grant PI 154/9-3). The author would like to thank the reviewers for their comments and suggestions.

References

Asher, N. and Lascarides, A. (2003). *Logics of Conversation*. Cambridge University Press.

Baldridge, J., Asher, N., and Hunter, J. (2007). Annotation for and robust parsing of discourse structure on unrestricted text. *Zeitschrift für Sprachwissenschaft*, 26(2):213–239.

Baldridge, J. and Lascarides, A. (2005). Probabilistic head-driven parsing for discourse structure. In *Proceedings of the Ninth Conference on Computational Natural Language Learning CoNLL-2005*, pages 96–103.

Blair-Goldensohn, S., McKeown, K. R., and Rambow, O. (2007). Building and refining rhetorical-semantic relation models. In *Proceedings of NAACL-HLT*.

Briscoe, T., Carroll, J., and Watson, R. (2006). The second release of the RASP system. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*.

Carletta, J. (1996). Assessing agreement on classification tasks: the kappa statistic. *Computational Linguistics*, 22(2):249–254.

Carlson, L., Marcu, D., and Okurowski, M. E. (2002). RST Discourse Treebank. Linguistic Data Consortium.

Clark, S. and Weir, D. (2000). A class-based probabilistic approach to structural disambiguation. In *Proceedings of the 18th International Conference on Computational Linguistics (Coling-00)*, pages 194–200.

Clark, S. and Weir, D. (2002). Class-based probability estimation using a semantic hierarchy. *Computational Linguistics*, 28(2).

Eugenio, B. D., Moore, J. D., and Paolucci, M. (1997). Learning features that predict cue usage. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*.

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

Forbes, K., Miltsakaki, E., Prasad, R., Sarkar, A., Joshi, A., and Webber, B. (2001). D-LTAG System – discourse parsing with a lexicalized tree adjoining grammar. In *Proceedings of the ESSLLI-01 Workshop on Information Structure, Discourse Structure and Discourse Semantics*.

Giampiccolo, D., Magnini, B., Dagan, I., and Dolan, B. (2007). The third PASCAL recognizing textual entailment challenge. In *Proc. of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 1–9.

Hickl, A. and Bensley, J. (2007). A discourse commitment-based framework for recognizing textual entailment. In *The Third PASCAL Recognizing Textual Entailment Challenge*, pages 171–176.

Le Thanh, H., Abeysinghe, G., and Huyck, C. (2004). Generating discourse structures for written text. In *Proceedings of COLING-04*, pages 329–335.

Li, H. and Abe, N. (1998). Generalizing case frames using a thesaurus and the MDL priciple. *Computational Linguistics*, 24(2):217–244.

Mann, W. C. and Thompson, S. A. (1987). Rhetorical structure theory: A theory of text organization. Technical Report ISI/RS-87-190, ISI, Los Angeles, CA.

Marcu, D. (2000). The rhetorical parsing of unrestricted texts: A surface-based approach. *Computational Linguistics*, 26(3):395–448.

Marcu, D. and Echihabi, A. (2002). An unsupervised approach to recognizing discourse relations. In *Proceedings of ACL-02*, pages 368–375.

Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. J. (1990). Introduction to WordNet: An on-line lexical database. *International Journal of Lexicography*, 3(4):235–312.

Murray, G., Renals, S., and Taboada, M. (2006). Prosodic correlates of rhetorical relations. In *Proceedings of HLT/NAACL ACTS Workshop*.

Nastase, V., Sayyad-Shiarabad, J., Sokolova, M., and Szpakowich, S. (2006). Learning noun-modifier semantic relations with corpus-based and WordNet-based features. In *Proceedings of AAAI*.

Pardo, T. A. S., das Graşas Volpe Nunes, M., and Rino, L. H. M. (2004). DiZer: An automatic discourse analyzer for brazilian portuguese. In *Proceedings of the 17th Brasilian Symposium on Artificial Intelligence (SBIA)*.

Pedersen, T., Banerjee, S., and Patwardhan, S. (2005). Maximizing semantic relatedness to perform word sense disambiguation. Research Report UMSI 2005/25, University of Minnesota Supercomputing Institute.

Polanyi, L., Culy, C., van den Berg, M., Thione, G. L., and Ahn, D. (2004). Sentential structure and discourse parsing. In *Proceedings of the ACL-04 Workshop on Discourse Annotation*.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, 14:130–137.

Redeker, G. (1990). Ideational and pragmatic markers of discourse structure. *Journal of Pragmatics*, 14:367–381.

Resnik, P. (1998). WordNet and class-based probabilities. In Fellbaum, C., editor, *WordNet: An Electronic Lexical Database*, pages 239–263. MIT Press.

Schapire, R. E. and Singer, Y. (2000). BoosTexter: A boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168.

Soricut, R. and Marcu, D. (2003). Sentence level discourse parsing using syntactic and lexical information. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*.

Sporleder, C. and Lascarides, A. (2005). Exploiting linguistic cues to classify rhetorical relations. In *Proceedings of Recent Advances in Natural Language Processing (RANLP-05)*.

Sporleder, C. and Lascarides, A. (2008). Using automatically labelled examples to classify rhetorical relations. *Natural Language Engineering*, 14(3):369–416.

Irene Cramer, Marc Finthammer, Alexander Kurek, Lukas Sowa, Melina Wachtling, Tobias Claas

Experiments on Lexical Chaining for German Corpora: Annotation, Extraction, and Application

1 Motivation

Converting linear text documents into documents publishable in a hypertext environment is a complex task requiring methods for segmentation, reorganization, and linking. The HyTex project, funded by the German Research Foundation (DFG), aims at the development of conversion strategies based on text-grammatical features. One focus of our work is on topic-based linking strategies using lexical chains, which can be regarded as partial text representations and form the basis of calculating topic views, an example of which is shown in Figure 1. This paper discusses the development of our lexical chainer, called GLexi, as well as several experiments on two aspects: Firstly, the manual annotation of lexical chains in German corpora of specialized text; secondly, the construction of topic views.

The principle of lexical chaining is based on the concept of lexical cohesion as described by Halliday and Hasan (1976). Morris and Hirst (1991) as well as Hirst and St-Onge (1998) developed a method of automatically calculating lexical chains by drawing on a thesaurus or word net. This method employs information on semantic relations between pairs of words as a connector, i.e. classical lexical semantic relations such as synonymy and hypernymy as well as complex combinations of these. Typically, the relations are calculated using a lexical semantic resource such as Princeton WordNet (e.g. Hirst and St-Onge (1998)), Roget's thesaurus (e.g. Morris and Hirst (1991)) or GermaNet (e.g. Mehler (2005) as well as Gurevych and Nahnsen (2005)). Hitherto, lexical chains have been successfully employed for various NLP-applications, such as text summarization (e.g. Barzilay and Elhadad (1997)), malapropism recognition (e.g. Hirst and St-Onge (1998)), automatic hyperlink generation (e.g. Green (1999)), question answering (e.g. Novischi and Moldovan (2006)), topic detection/topic tracking (e.g. Carthy (2004)).

In order to formally evaluate the performance of a lexical chaining system in terms of precision and recall, a (preferably standardized and freely available) test set would be required. To our knowledge such a resource does not yet exist–neither for English nor for German. Therefore, we conducted several annotation experiments, which we intended to use for the evaluation of GLexi. These experiments are summarized in Section 2 . The findings derived from our annotation experiments also led us to developing the highly modularized system architecture, shown in Figure 4, which provides interfaces in order to be able to integrate different pre-processing steps, semantic relatedness measures, resources and modules for the display of results. A survey of the architecture and the

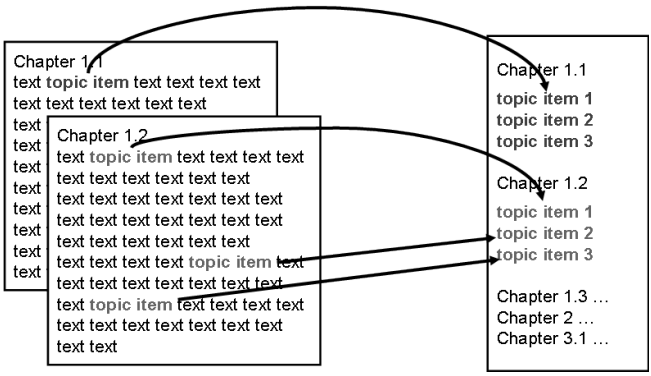


Figure 1: Construction of a Topic View Based on a Selection of Topic Items (= Thematically Central Lexical Unit) per Paragraph

single modules is provided in Section 3. The challenges we experienced while annotating lexical chains brought us to analyze the performance of GLexi by means of a multi-level evaluation procedure, which is discussed in Section 4.

2 Annotation Experiments

The annotation experiments referred to below were originally intended to facilitate the development of annotation guidelines and thereby to promote the formulation of a gold standard for the evaluation of GLexi. However, the results of a preliminary study as well as the experiments detailed in the literature on English data (see, among others, Morris and Hirst (2005) as well as Beigman Klebanov (2005)) demonstrate that a satisfactory degree of inter-annotator agreement is not yet achieved in the manual annotation of lexical chains .

From our point of view, this is due to at least three aspects: Firstly, the subjects are focussed on building an individual understanding of the text, which obscures the various features that establish text cohesion, such as lexical cohesion or deixis. Secondly, the subjects also appear to struggle with differentiating between different features of textual cohesion. Particularly the anaphora and coreference resolution appears to be interacting strongly with lexical cohesion and thus the lexical chains in a text (see e.g. Stührenberg et al. (2007)). Thirdly, there is no consensus among researchers with respect to the semantic relations relevant in lexical chaining. It was therefore impossible to ensure a consistent annotation in regard to the relation types considered.

For this reason, all three experiments described in the following should be regarded as pilot studies. They were drafted and conducted with the aim of gaining more knowledge on lexical chaining. We deemed as particularly important, which aspects of computing lexical chains or of their manual annotation respectively might be relevant for our application scenarios, namely, the construction of topic views. Contrastingly, it was of less importance, whether a satisfactory inter-annotator agreement could be achieved.

Therefore, the actual evaluation of GLexi was not conducted by means of the data, which were annotated in the experiments, but is rather based on an evaluation procedure that is detailed in Cramer and Finthammer (2008a) and sketched in Section 4. In altogether all three annotation experiments, we had subjects annotate lexical chains or pre-stages/parts of lexical chains within texts. The task of the three experiments may be summarized as follows:

- experiment 1: manual annotation of lexical chains;
- experiment 2: manual search for (direct and indirect) relations between words or synsets within GermaNet;
- experiment 3: manual annotation of lexical chains, represented as mind-maps.

In **experiment 1**, seven subjects (all subjects were second-year students of philology or linguistics with no background in lexicography and no knowledge in applied or computational linguistics) were asked to annotate lexical chains within three texts (two newspaper/ magazine articles, one from faz.net and unicum.de respectively, as well as a lexicon entry out of the German Wikipedia). For this purpose, the subjects were given a 15-minute oral introduction to the concepts of lexical cohesion and lexical chaining, including some notes on the theoretical background as described by Halliday and Hasan (1976) and some example chains. Subsequently, they had five minutes to ask clarification questions. The subjects were then given the following documents (partially depicted in Figure 2): a list of all nouns of the three texts, an evaluation questionnaire (for evaluating the relevance of the noun or phrase for their text comprehension), a template for generating the chains, a list of the relations to be considered, and a feedback questionnaire. Thereupon, the subjects were asked to complete the task as far as possible within one hour. In order to get an impression of the time necessary to annotate a certain amount of text we limited the amount of time.

Results - experiment 1: Nearly all subjects aborted the annotation before the set time exceeded. In fact, the subjects found the evaluation of the relevance of a noun to be comparatively easy, while they found the actual annotation of lexical chains to be rather difficult. Based on their divergent solution strategies in annotating lexical chains, the subjects may be subsumed into two groups (with three or four subjects each): the first group reinterpreted the task in so far, as they organized the nouns in nets (which they themselves called mind maps) rather than in chains. Subjects of the second group changed their strategies of chaining several times throughout the set time and in doing so crossed out previous versions in order to substitute them with improved ones (e.g. versions containing more or less entries or versions connected via other relations).

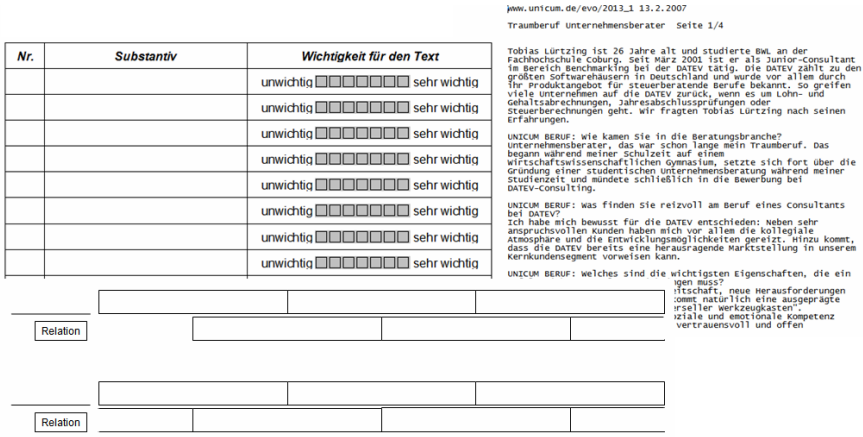


Figure 2: Documents of Experiment 1: Questionnaire for the Evaluation of the Relevance for the Text, Text Extract, Template for the Manual Annotation of Lexical Chains, e.g. in the Displayed Text Extract

By means of a subsequent oral interview as well as the feedback questionnaire, we were able to identify the following main aspects of our subjects' criticism, which referred to the set-up of the experiment, and particularly to the annotation of lexical chains. The majority of the subjects were of the following opinions:

- a linearization of the annotation (that is generating chains) is impossible, or at least not sensible or functional respectively;
- constructing nets instead of chains might be more appropriate to the task;
- many nouns could not be incorporated into the chains/nets, as they either were of too little thematic relevance for the text or they would have been better considered as a phrase;
- the relations accounted for in the experimental setup were not sufficient.

These results are congruent with the experiments on the annotation of lexical chains for English data by Beigman Klebanov (2005) as well as Morris and Hirst (2005) respectively. The following consequences drawn from experiment 1 led to the experimental set-ups for both subsequent experiments. Firstly, it seemed functional to segment the annotation into subtasks; furthermore, the obligatory linearization was to be substituted by net structures;

and finally, the subjects were to be allowed to consider larger phrases instead of nouns only.

As a consequence, in **experiment 2** three subjects (of the initial seven) were asked to trace direct or indirect relations respectively in GermaNet for a list of word pairs, (thus the subjects were asked to find complex combinations of relations, in short paths, i.e. the path between `Blume` (Engl. flower) and `Baum` (Engl. tree), which spans three steps in regard to GermaNet, namely hypernymy - hyponymy - hyponymy) by means of a graphic user interface for GermaNet (see Finthammer and Cramer (2008) for more information on this). We thus intended to account for the complaints by our subjects in experiment 1 that the semantic relation types did not suffice in order to satisfactorily complete the annotation of lexical chains. The subjects were given a fraction of the word pairs of experiment 1 and were asked to trace paths between these words with respect to GermaNet; they had a time-frame of four hours to complete the task.

Results - experiment 2: In principle, the following four constellations (see Cramer and Finthammer (2008b) for examples) could be identified:

- intuitively, a **semantic relation exists** between the words of the pair and this **connection can easily be identified** within GermaNet;
- intuitively, a **semantic relation exists** between the words of the pair, **but this relation can not easily or not at all be identified** within GermaNet;
- Intuitively, **no semantic relation exists** between the words of the pair, **but a short path** can easily be identified between the words within GermaNet;
- intuitively, **no semantic relation exists** between the words of the pair, **and no short path nor a path at all** can be identified within GermaNet.

In spite of the graphic user interface (see Finthammer and Cramer (2008)), there were almost no cases where the subjects were able to identify a path in GermaNet within an acceptable time-frame. In most cases, the search for a path terminated after two or three steps without any results. In these cases, the subjects were not able to decide intuitively on the next steps. Admittedly, it is not surprising that paths can only be detected manually with a great expenditure of time. But the results, that on the one hand even short paths run across inappropriate nodes (also see Cramer and Finthammer (2008b)) and that, on the other hand, intuitively, nodes being close to each other are only connected via long paths are markedly critical for the qualitative evaluation of word-net based relatedness measures.

In **experiment 3**, two subjects (of the above mentioned seven) were asked to construct lexical nets (similar to mind-maps) for three texts on the basis of the concept of lexical cohesion. We instructed them to consider the introduction they were given in experiment 1 as well as the results of the oral interviews and the feedback questionnaires. They first segmented the texts into paragraphs, for each of which one net was to be created. In a next step, the words and phrases of the paragraphs were transferred into net structures, which may be regarded as a partial representation of the textual content. An example of

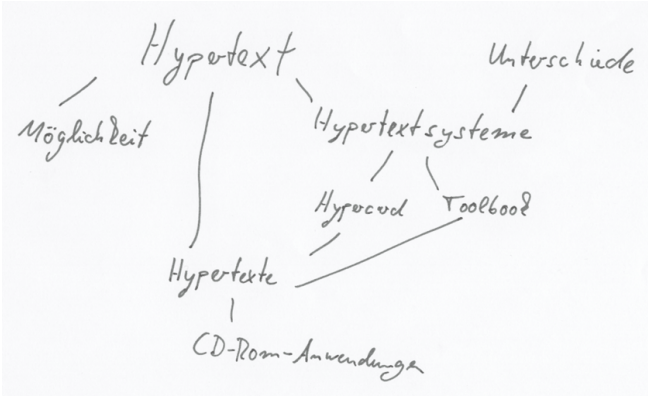


Figure 3: Example of a Manually Generated Net Structure as Complement or Substitute for Lexical Chains

this can be found in Figure 3. As the figure illustrates, independently of each other, both subjects organized the words and phrases of the respective paragraph departing from a center (in regard to content).

The **results of the three annotation experiments** can be summarized as follows: The annotation of lexical chains within texts forms a complex task and is hardly viable along with achieving a sufficiently strong agreement of the subjects. These results correspond–in our opinion–to the results for English data as described in Beigman Klebanov (2005) as well as Morris and Hirst (2005). In a nutshell, developing sustainable annotation guidelines from the different experiments was ultimately impossible. Nevertheless, the results were relevant for our subsequent research on lexical chains for German data:

- Firstly, representing lexical chains as nets led us to the idea that the lexical units of a paragraph might be arranged around one or more words/word groups and thus around one or more thematic center(s) (we call them topic items). These topic items seem to feature a dense net structure and strong relations, which, in turn, forms the basis for the construction of topic views.

- Apart from this, the results emphasize that the performance of a system for calculating lexical chains cannot be evaluated by means of a manually annotated data. For this reason, an alternative approach for evaluation needed to be designed. Our suggestion for such an alternative procedure is sketched in Cramer (2008) and is briefly outlined in Section 4.

Table 1: Options of Parameterization of GLexi Including a Compilation of the Configurations Used so far in the Experiments

Adjustable Parameters	Used Parameters
Pre-Processing: sentence boundary detection, tokenization, POS-tagging, morphological analysis, chunking	all pre-processing steps
Resources: GermaNet, GermaTermNet (see Beißwenger (2006) for more information on GermaTermNet), Google-API	GermaNet and GermaTermNet
Relatedness Measures: 8 based on GermaNet 3 based on Google (see e.g. Cilibrasi and Vitanyi (2007))	all 8 GermaNet based measures

3 GLexi–Architecture

Drawing on the results of the previously described annotation experiments, we devised a modular system for calculating lexical chains/nets within German corpora. The basic modules of our system called GLexi (spoken: `galaxy`) are summarized in Figure 4. All modules are designed in such a way that the user of GLexi is able to additionally integrate own components, such as alternative pre-processing steps or resources. All options of parameterization–which were subject to our experiments up to now and which we use for calculating topic items (and topic views)–are compiled in Table 1.

The depiction of the system structure in Figure 4 also illustrates the chaining procedure: Based on the input (in an XML format particularly devised for this purpose) GLexi initially checks which units are to be considered as chaining candidates. Thereafter, all information on the candidates contained in the input is collected and hence is available for the core algorithm as well as the output generation. For each candidate pair GLexi then tests whether a suitable semantic relation can be identified on the basis of the selected resource and semantic relatedness measure. If this is the case, the pair is considered as a chaining pair and accordingly stored in the meta-chains[1] including its relatedness measure value. Having calculated the relatedness measure values for all possible pairs, i.e. having filled the meta-chains, the output of the results can be constructed. Again, different options are available: apart from the actual lexical chains (see e.g. the algorithm by Hirst and St-Onge (1998)) it is also possible to display all candidates including their relations as a net structure. An example of this is depicted in Figure 5. Obviously, we derived this format from the net structures as they were manually generated by our subjects in the annotation experiment 3 (see Section 2 and Figure 3). The net structure

[1] See Silber and McCoy (2002) for more information on the concept of meta-chains.

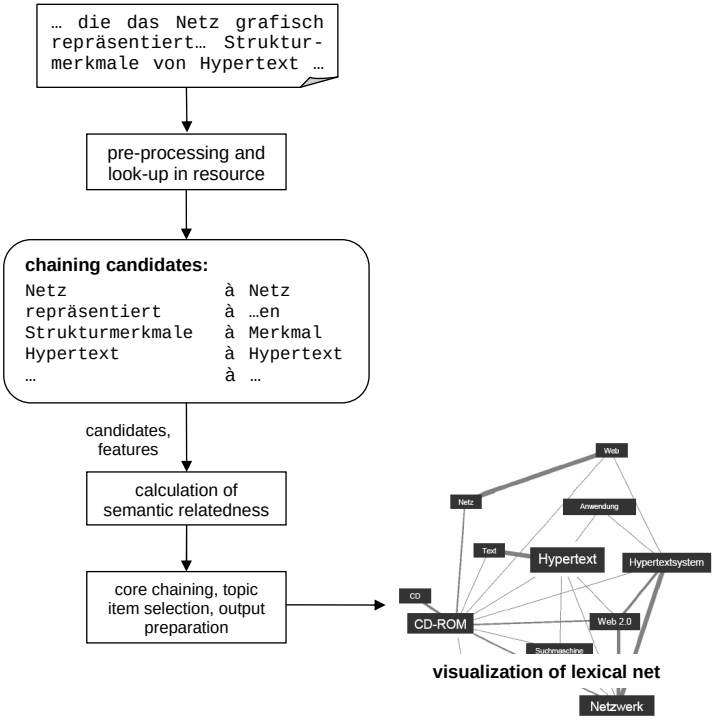


Figure 4: Architecture of GLexi

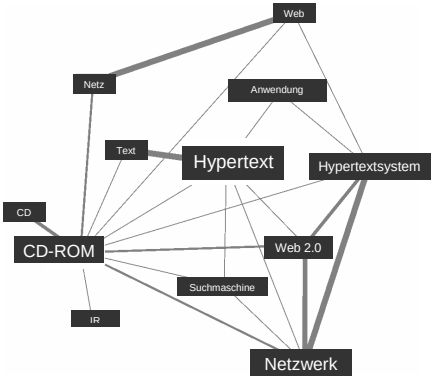


Figure 5: Example of a Lexical Net Generated by Means of GLexi

as a substitute for classical lexical chains, also forms the basis for calculating topic items (and topic views), as depicted in Figure 6.

4 GLexi–Evaluation

As mentioned above, no gold standard has been compiled so far for the evaluation of a lexical chainer and, in addition, the previously described results of the experiments illustrate that the manual annotation of such a gold standard represents yet unsolved challenges. We therefore suggest a four-step evaluation procedure as an alternative approach. A detailed discussion of this evaluation procedure is provided in Cramer and Finthammer (2008a). Therefore we limit the following description of the procedure to aspects relevant to the computation of topic views.

For evaluating GLexi, we drew on GermaNet (see e.g. Lemnitzer and Kunze (2002), version 5.0), the Google-API, and a word frequency list provided by S. Schulte im Walde[2] as resources for our eleven semantic relatedness measures. We additionally used parts of the HyTex core corpus (see Beißwenger and Wellinghoff (2006)), which we pre-processed by means of the Temis Tools[3] and transformed into the previously mentioned XML format.

[2]We kindly thank Dr. Schulte im Walde for her support.

[3]For the experiments described here, the Insight DiscovererExtractor Version 2.1 was used. We also kindly thank the Temis group for supplying their software and for their support.

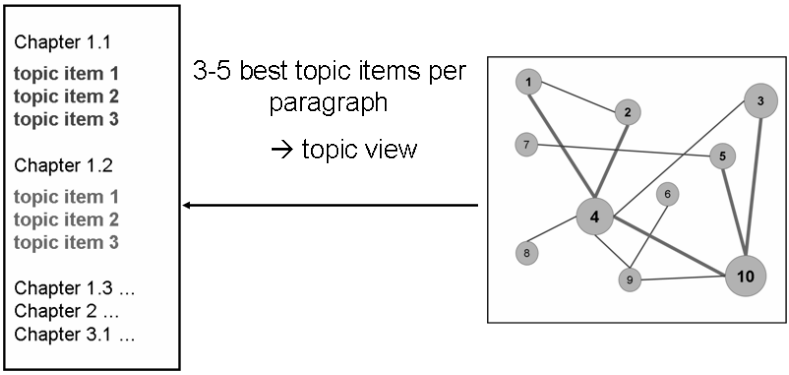


Figure 6: Output of GLexi as a Lexical Net Forms Basis for Calculating Topic Items and Topic Views: Choose the 3-5 Most Important Topic Items, Supplement TOC Accordingly.

Table 2: Coverage of GermaNet in Regard to the HyTex Core Corpus

Approx. 29,000 (Noun) Tokens split into			
56% in GermaNet	44% not in GermaNet		
	15% inflected	12% compounds	17% proper names nominalization, abbreviation etc.

4.1 Coverage

With respect to the coverage of GLexi, two settings may be distinguished according to the resource used: If GermaNet forms the basis for calculating lexical chains, approximately 56% of the noun tokens in our corpus will be covered, see Table 2. If, in turn, the calculation of the semantic relatedness is based on the co-occurrence measures based on the Google-API, all words in the texts are accounted for. Having said that, using Google based relatedness measures involves two essential shortcomings: firstly, it does not provide a word sense disambiguation on-the-fly, as is the case using e.g. GermaNet; secondly, as the results given in Section 4.3 demonstrate, the correlation between the Google co-occurrence based measures and the average assessments of the subjects in regard to semantic relatedness still ranges below the measures which were achieved using the measures based on GermaNet.

Table 3: Correlation between Human Judgements and Relatedness Measure Values with Respect to the 100 Word Pairs

	Graph Path	Tree Path	Wu-Palmer	Leacock-Chodorow
correl.	0,41	0,42	0,36	0,48
	Hirst-StOnge	Resnik	Jiang-Conrath	Lin
correl.	0,47	0,44	0,45	0,48
	Google Quotient	Google NGD	Google PMI	
correl.	0,24	0,29	0,27	

4.2 Quality of Disambiguation

In order to evaluate the quality of word sense disambiguation, we manually annotated a fraction of the HyTex core corpus. As a next step, lexical chains were calculated for these data; in deciding upon the affiliation of a word (or a lexical unit of the corpus respectively) with a lexical chain, its word sense is simultaneously disambiguated. By comparing the decisions made on the basis of the chains calculated with the manual annotation, the quality of the disambiguation of GLexi may be assessed. Depending on the measure used, the results range between approximately 35% and 60%. In regard to the quality of their disambiguation, the measures introduced by Resnik (1995), Wu and Palmer (1994) and Lin (1998) perform best.

4.3 Quality of Semantic Relatedness Measures

In order to evaluate the performance of our eleven relatedness measures, we drew on a method typically employed in this context, namely, we compared the semantic relatedness measures values for a list of word pairs with human judgements of these pairs. Thus, the average assessments and the associated automatically calculated relatedness measure values for the word pairs are juxtaposed: Table 3 depicts the correlation between the human judgements and the eleven measures. Obviously, the measure values are scattered, which results in the rather low correlation coefficients. A detailed analysis of our human judgement experiments and a comparison with similar studies can be found in Cramer and Finthammer (2008a) and Cramer (2008).

4.4 Application Scenario

As mentioned above, the application scenario we aim at is the construction of topic views, an example of which is displayed in Figure 1. In order to automatically calculate topic views for given text passages, we mainly draw on the lexical nets generated by GLexi. We

integrated the (in the following described) algorithm for the calculation of topic views as an additional module of GLexi: on the basis of the lexical nets we rank the words/phrases (topic item candidates) of a passage with respect to their topic strength; thus, we rank the candidates which are most relevant at the top of a topic item candidate list. The decision on the ranking of a given topic item candidate is mainly based on three feature types: firstly, the density of the lexical net for the given candidate, secondly, the strength of its relations, and, thirdly, its tf/idf score. We regard the top three to five (depending on the length of the passage) topic item candidates as the topic items of the given passage and construct the topic view by supplementing the topic items to the table of contents. In order to evaluate the performance of the above described algorithm, we drew on the manual annotation of topic items. Initial annotation experiments show that an inter-annotator agreement of approximately 65% can be achieved. We also found that when evaluating the automatic calculation of topic views with respect to the manual annotated data, an overlap of 55% to 75% can be achieved. Our initial results also stress that GLexi is able to compute high quality topic views if the passages are of a certain length and if the topic item candidates are appropriately modeled in the lexical semantic resource employed. Interestingly, in spite of the moderate performance of GLexi with respect to its coverage, its word sense disambiguation performance and the semantic relatedness measures used, we were able to achieve–with only a few simple features–relatively good results in calculating topic views. However, we certainly need to systematically explore the calculation of topic views in a follow-up study.

5 Outlook

The results of our annotation experiment describe here as well as the evaluation of our system GLexi demonstrate that the concept of lexical chains as well as their automatic construction leaves a number of aspects unsettled: Firstly, it is questionable to what extent lexical chains may be distinguished from anaphoric structures or coreference respectively, or, put vice versa, how far these three concepts might be merged into a homogenic concept. Moreover, it remains unclear, whether we are dealing with lexical nets rather than lexical chains–as the subjects of experiment 1 stressed. The experiments on the construction of topic views however show that it might indeed be reasonable to replace the concept of lexical chains by a new concept of lexical nets. We therefore plan, as a follow-up study, to investigate the (basic) features of lexical nets and also intend to incorporate the findings of linguists on lexical (semantic) cohesion into this new concept more thoroughly. Secondly, the moderate performance of GLexi as detailed in Sections 4.1 to 4.3 indicates that lexical chaining (or lexical netting) might be a not yet well understood method for the construction of partial text representations. We find that particularly the quality of word sense disambiguation (which should–at least according to the theory of lexical chaining–be conducted on-the-fly while chaining words of a text) and the performance of the semantic relatedness measures do not meet our demands. The quality of disambiguation might well be improved by enhancing the pre-processing, but still the problem of calculating the semantic relatedness remains unsettled. The

latter, again, consists of diverse sub-aspects: First of all, although there has been much research (see Morris and Hirst (2004) as well as Boyd-Graber et al. (2006)) on the question which types of semantic relations are actually relevant at all (for the calculation of lexical chains as well as in principle), we consider this issue unsettled. In addition, the human judgement experiments typically used in order to asses the performance of a semantic relatedness measure, are–in our opinion–not well understood, i.e. it is unclear what exactly is measured in such an experiment, and furthermore, the experimental set-up is not well defined. And finally, all measures which have been taken into account so far–do not consider those relations that arise exclusively from the content of the text and which can evolve within a text only. Despite these numerous unsettled questions, the first application-based results demonstrate that lexical chains are convenient and helpful for the calculation of topic items and topic views. We therefore intend to systematically investigate–which parameter settings perform best for the calculation of topic views–and feel confident that we will–in the long run–be able to achieve results of high quality for our corpora of specialized text.

Acknowledgement

We kindly thank Angelika Storrer, Michael Beißwenger, Christiane Fellbaum, Claudia Kunze, Lothar Lemnitzer, Alexander Mehler, and Sabine Schulte im Walde for their support and valuable comments and suggestions. Moreover, we thank our dedicated subjects.

References

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

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

Beißwenger, M. (2006). Termnet–ein terminologisches Wortnetz im Stile des Princeton Wordnet. Technical report, University of Dortmund, Germany.

Beißwenger, M. and Wellinghoff, S. (2006). Inhalt und Zusammensetzung des Fachtextkorpus. Technical report, University of Dortmund, Germany.

Boyd-Graber, J., Fellbaum, C., Osherson, D., and Schapire, R. (2006). Adding dense, weighted, connections to wordnet. In *Proceedings of the 3rd Global WordNet Meeting*, pages 29–35.

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

Cilibrasi, R. and Vitanyi, P. M. B. (2007). The google similarity distance. *IEEE Transactions on Knowledge and Data Engineering*, 19.

Cramer, I. (2008). How well do semantic relatedness measures perform? a meta-study. In *Proceedings of the Symposium on Semantics in Systems for Text Processing*.

Cramer, I. and Finthammer, M. (2008a). An evaluation procedure for word net based lexical chaining: Methods and issues. In *Proceedings of the 4th Global WordNet Meeting*, pages 120–147.

Cramer, I. and Finthammer, M. (2008b). Tools for exploring germanet in the context of cl-teaching. In *Text Resources and Lexical Knowledge. Selected Papers from the 9th Conference on Natural Language Processing KONVENS 2008*.

Finthammer, M. and Cramer, I. (2008). Exploring and navigating: Tools for germanet. In *Proceedings of the 6th Language Resources and Evaluation Conference*.

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

Gurevych, I. and Nahnsen, T. (2005). Adapting lexical chaining to summarize conversational dialogues. In *Proc. of the Recent Advances in Natural Language Processing Conference (RANLP 2005)*.

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

Hirst, G. and St-Onge, D. (1998). Lexical chains as representation of context for the detection and correction malapropisms. In Fellbaum, C., editor, *WordNet: An electronic lexical database*.

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

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

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

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

Morris, J. and Hirst, G. (2004). Non-classical lexical semantic relations. In *Proc. of HLT-NAACL Workshop on Computational Lexical Semantics*.

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

Novischi, A. and Moldovan, D. (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*.

Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. In *Proc. of the IJCAI 1995*.

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

Stührenberg, M., Goecke, D., Diewald, N., Mehler, A., and Cramer, I. (2007). Web-based annotation of anaphoric relations and lexical chains. In *Proc. of the Linguistic Annotation Workshop, ACL 2007*.

Wu, Z. and Palmer, M. (1994). Verb semantics and lexical selection. In *Proc. of the 32nd Annual Meeting of the Association for Computational Linguistics*.

Maja Bärenfänger, Daniela Goecke, Mirco Hilbert, Harald Lüngen, Maik Stührenberg

Anaphora as an Indicator of Elaboration: A Corpus Study

This article describes an investigation of the relationship between anaphora and relational discourse structure, notably the ELABORATION relation known from theories like RST. A corpus was annotated on the levels of anaphoric structure and rhetorical structure. The statistical analysis of interrelations between the two annotation layers revealed correlations between specific subtypes of anaphora and ELABORATION, indicating that anaphora can function as a cue for ELABORATION.[1]

1 Introduction

Two aspects of the structure of discourse are *relational discourse structure* and *anaphoric structure*. There are two views regarding the relationship between these two levels of analysis: On the one hand, relational, hierarchical discourse structure is said to provide domains of accessibility for antecedent candidates of anaphoric expressions (Polanyi, 1988; Cristea et al., 2000; Asher and Lascarides, 2003). On the other hand, coreference plays a role in the definition of certain discourse relations, notably ELABORATION (Corston-Oliver, 1998; Carlson and Marcu, 2001; Knott et al., 2001), but also LIST e.g. in the discourse parsing approach by Corston-Oliver (1998, p. 137).

In an automated analysis of relational discourse structure of text, lexical discourse markers (i.e. conjunctions and sentence adverbials) play a major role as cues for identifying discourse relations (Marcu, 2000; Le Thanh et al., 2004). ELABORATION, however, is a discourse relation frequently not signalled by lexical discourse markers, hence the question arises whether one could systematically use anaphora as a cue for identifying ELABORATION. This study presents an empirical investigation of the relationship between discourse anaphora and relational discourse structure by means of an analysis of a text corpus that was annotated independently on these two levels of linguistic description. We focus on anaphoric structure as a cue for discourse structure, in particular, Elaboration. The remainder of the article is structured as follows: In Section 2, we provide the theoretical background of coreference and relational discourse structure as well as our categorial framework of anaphora and rhetorical relations and formulate our research questions in terms of these. In Section 3 we give an overview of our corpus of German scientific articles, the annotation schemes used for anaphora and rhetorical structure, and the methods used in querying and statistically analysing the corpus. In Section 4, the results of the corpus analysis are presented and discussed. In

[1] The work presented in this article is a joint effort of the projects A2 (*Sekimo*) and C1 (*SemDok*) of the Research Group 437 *Text-technological modelling of information* funded by the German Research Foundation DFG.

Section 5, we describe the implementation of some of our findings in a discourse parser, and present an evaluation of parsing experiments with and without anaphoric cues.[2]

2 Two aspects of discourse structure

2.1 Anaphora

Anaphoric relations as a cohesive device are an important factor of the coherence of texts. Anaphora occurs when the interpretation of a linguistic unit (the anaphor) is dependent on the interpretation of another element in the previous context (the antecedent). The anaphor is often an abbreviated or reformulated reference to its antecedent and thus provides for the progression of discourse topics. The analysis of anaphora as a device for discourse structure presupposes the notions of *discourse entities* and *discourse segments* (cf. Webber, 1988), the latter building the bridge to relational discourse structure.

Discourse entities – or discourse referents in the terminology of Karttunen (1976) – serve as constants within a discourse model which are evoked by (mainly) NPs and which can be referred to in the subsequent discourse. Following Webber (1988, p. 113), NPs can either evoke new discourse entities in the discourse model (or universe) or can "refer to ones that are already there". Pronouns do not evoke new discourse entities but access existing ones (cf. Webber, 1986). In DRT (Kamp and Reyle, 1993), a slightly different view on NPs evoking discourse referents is adopted. Each discourse is represented by a *discourse representation structure* (DRS), and each DRS consists of two components: a set of discourse referents (the *universe*) and a set of conditions. Both pronouns and NPs add discourse referents to the discourse universe and anaphoric relations to already existing referents are modelled via identity assertions whereas according to Webber (1986, 1988) an anaphoric relation holds directly by accessing already existing discourse entities.

For the investigation described in this article nominal discourse entities have been introduced for pronouns as well as for definite and indefinite NPs and anaphoric relations have been annotated manually on the basis of the discourse entities. Apart from anaphoric relations with antecedents of nominal type, anaphoric elements may also refer to antecedents that have been evoked by non-nominal units. Asher (1993, p. 35) uses the term *abstract entity anaphora* where "not just sentential nominals but other constructions like verb phrases or even whole sentences introduce abstract objects and eventualities into a discourse and may serve as referents for anaphoric pronouns".[3]

The following examples with nominal (1), sentential nominal (2), and verb phrase antecedents (3) illustrate the distinction between nominal and non-nominal discourse entities.

[2]We would like to thank the two anonymous reviewers who provided valuable comments on a previous version of this article.

[3]The term *sentential nominal* refers to constructions that are semantically related to sentential structures, e.g. due to a derived nominal as in Example (2) (cf. Asher, 1993).

(1) I met a man yesterday. He told me a story.
(Example taken from Clark, 1977, p. 414)

(2) [The destruction of the city]$_i$ amazed Fred. It$_i$ had been bloody.
(Example taken from Asher, 1993, p. 35)

(3) John saw [Mary cross the finish line first in the marathon]$_i$. Two days later, he still didn't believe it$_i$. (Example taken from Asher, 1993, p. 39)

The term discourse segment refers to either elementary spans of texts (clauses, sentences and the like) or complex segments that are built up recursively from elementary segments. Discourse segments and relations between them form the discourse structure which is of special interest for discourse anaphora; the interrelationship between anaphora and discourse structure is manifested in several approaches to discourse structure: Intentional approaches like Centering Theory (Grosz and Sidner, 1986; Grosz et al., 1995) model anaphora according to different relations between adjacent scentences. Informational approaches like SDRT (Segmented Discourse Representation Theory, Asher and Lascarides, 2003) model anaphoric relations on the basis of accessibility according to the underlying discourse structure. Discourse structure as a constraint for anaphoric relations is prominent in the *Right Frontier Constraint* (Polanyi, 1988). Furthermore, application-oriented approaches (e.g. Cristea et al., 1998, 2000) focus on the detection of appropriate antecedent candidates within an anaphora resolution system and use discourse structure as a constraint for anaphoric relations.

For a description of anaphoric relations one has to differentiate between the linguistic form of text spans between which anaphoric relations hold on the one hand and the semantic interpretations of the respective text spans, i.e. the discourse entities, on the other hand.

A taxonomy according to the linguistic form of the anaphoric element classifies anaphora into nominal anaphora, verb anaphora, adverb anaphora, zero anaphora and the like. Furthermore, the antecedent for nominal anaphors may be of nominal type or a non-nominal construction that refers to an abstract entity (e.g. events, facts, propositions; cf. Asher, 1993).

According to the relations that hold between the discourse entities, anaphora can be further divided into direct anaphora and indirect anaphora. For direct anaphora, the antecedent is explicitly mentioned in the previous context (Example (1) above) whereas for indirect anaphora the antecedent is not mentioned explicitly but has to be inferred from the context (Example (4)).

(4) I looked into the room. The ceiling was very high.
(Example taken from Clark, 1977, p. 415)

The latter is also referred to as *bridging relations* following the terminology of Clark (1977). Apart from the distinction of direct/indirect anaphora, discourse referents may be coreferent or not. In Example (1) the linguistic units "a man" and "he" are co-specified

and refer to the same entity whereas "the room" and "the ceiling" in Example (4) do not although they are closely related due to world knowledge.

The distinction of anaphora according (a) to the linguistic form of anaphor and antecedent and (b) to the relations that hold between anaphor and antecedent leads to a taxonomy of anaphoric relations consisting of two primary relations which can be used for a broad annotation and two sets of secondary relation types for a more fine-grained annotation. This taxonomy forms the basis for the annotation of anaphoric relations and has been defined, together with the annotation scheme, on the basis of Holler-Feldhaus (2004) and Holler et al. (2004). The annotation scheme is described in detail in Goecke et al. (2007) and in (Diewald et al., 2008, this volume). The primary relation types (cospecLink and bridgingLink) allow for a distinction of direct and indirect anaphora and may be further subdivided into secondary relation types according to the relation between anaphor and antecedent (see Figure 1).

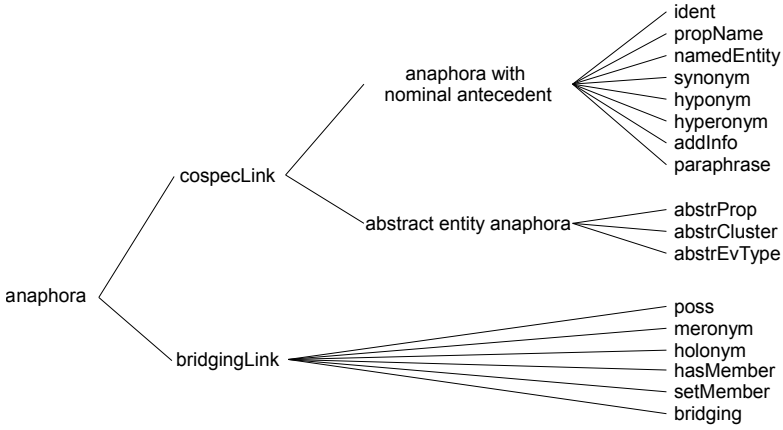


Figure 1: Sekimo hierarchy of anaphoric relations

For cospecLink two sets of secondary relations exist: one set for relations with antecedents of nominal type and one set for abstract entity anaphora. The subtypes of abstract entity anaphora are characterised as follows: abstrProp describes anaphoric relations with an antecedent of propositional type, abstEvType describes anaphoric relations with an event type antecedent, and abstrCluster describes anaphoric relations where the anaphor refers to a cluster of propositions. For nominal antecedents, we annotate eight secondary relation types: The relation ident is chosen for pronominal anaphors or anaphor-antecedent pairs with identical head noun. The value propName is chosen if the anaphoric element is a proper name that refers to an NP antecedent.

Anaphors that are not of type NAMEDENTITY but refer to an antecedent of type NAMEDENTITY are annotated with the respective relation type. Synonymy between the head nouns of anaphor and antecedent is annotated using the value SYNONYM. HYPERONYMY and HYPONYMY are chosen accordingly. The values ADDINFO and PARAPHRASE are chosen if the anaphor adds new information to the discourse or if the anaphor is a paraphrase of its antecedent.

For bridging relations six secondary relation types have been defined: The value POSS describes a possession relation between the anaphor and its antecedent. The value MERONYM is chosen in case of a part-whole-relation between the head nouns of anaphora and antecedent; HOLONYM is chosen accordingly. The value HASMEMBER is chosen if the anaphor describes a set and the antecedent(s) are part of of that set and SETMEMBER is chosen if the anaphoric elements is part of a set described by its antecedent. If none of the previous relation types hold the relation is annotated using the value BRIDGING.

The taxonomy shows that not only pronominal anaphors or definite descriptions with identical head nouns are taken into account for the investigation of anaphora and relational discourse structure. The majority of the relation types are relevant for definite description anaphors whose relations are licensed by lexical-semantic relations or association (e.g. *birthday party - presents*). Both intra- as well as inter-sentential anaphora is taken into account; definite description anaphors tend to find their antecedents across sentence boundaries even at a large distance between anaphor and antecedent. Consequently, anaphor and antecedet are frequently located within different discourse segments, allowing for an an investigation of the relationship between discourse anaphora and relational discourse structure.

The applicability of the taxonomy for corpus annotations has been tested in a study on inter-annotator agreement. The results of the study show that annotators are able to annotate even fine-grained secondary relation types reliably (cf. Goecke et al., 2008).

2.2 Rhetorical structure

Relational discourse structure is covered by several linguistic theories of discourse like SDRT, the Unified Linguistic Discourse Model (ULDM, Polanyi, 1988; Polanyi et al., 2003), or Rhetorical Structure Theory (RST, Mann and Thompson, 1988; Marcu, 2000). In the framework of RST, which we focus on here, discourse structure consists of relationally connected discourse segments which can be either elementary or complex. Segments are combined to form larger segments by two types of discourse relations: mononuclear or multinuclear relations. In a mononuclear relation, one discourse segment has the status of a "nucleus" (N), the more "essential" piece of text, the other segment has the status of a "satellite" (S), a less essential text part "more suitable for substitution" (cf. Mann and Thompson, 1988). In a multinuclear relation, all related segments serve as nuclei. The original RST distinguishes 26 mono- or multinuclear relations; like other projects (cf. Carlson et al., 2001; Hovy and Maier, 1995), we extended this relation set

with subrelations according to requirements of our corpus and application scenario (cf. Lüngen et al., 2006).

One prominent relation in our corpus is the mononuclear relation ELABORATION. Mann and Thompson (1988) introduced ELABORATION into RST by defining conditions on the combination of two discourse segments S and N for ELABORATION to hold:

> S presents additional detail about the situation or some element of subject matter which is presented in N or inferentially accessible in N in one or more of the ways listed below. In the list, if N presents the first member of any pair, then S includes the second:
> 1. set:member
> 2. abstract:instance
> 3. whole:part
> 4. process:step
> 5. object:attribute
> 6. generalization:specific
>
> (*ibid.* p. 273).

The relations enumerated in this listing partly resemble the semantic relations introduced in Section 2.1. The use of "presents", "presented in" and "includes" in the definition suggests that the relations listed are supposed to hold between entities that are in a sense contained in the segments N and S.

Corston-Oliver (1998, p. 81), who focuses on discourse parsing, argues that ELABORATION is amongst other things indicated by "subject continuity" which he describes as being "the most important kind of referential continuity for identifying discourse relations". In his "worked example" (*ibid.* p. 203f), cf. Example (5), subject continuity is clearly realised by the anaphoric pronoun *it*, and subject continuity also appears in his list of cues for ELABORATION (*ibid.* p. 103).

(5) [The aardwolf is classified as Proteles cristatus]$_{Nuc}$. [It is usually placed in the hyena family, Hyaenidae. {...}]$_{Sat}$

(Example taken from Corston-Oliver, 1998, p. 203f; originally from an article in the *Microsoft Encarta 96* Encyclopedia)

Wolf and Gibson (2006, p. 32) also use an ELABORATION relation in their discourse annotation schema (which is not based on RST) and define it in their coding procedure as providing "more detail about an already introduced entity or event".

(6) [Crawford & Co., Atlanta (CFD) began trading today]$_{Nuc}$. [Crawford evaluates health care plans, manages medical and disability aspects of worker's compensation injuries and is involved in claims adjustments for insurance companies.]$_{Sat}$

(Coding example in Wolf and Gibson, 2006, p. 32f; originally from text wsj-0607 (Wall Street Journal Corpus) from Harman and Liberman (1993))

In their coding example (Example (6)), the discourse entity named *Crawford* is referred to by linguistic expressions in both segments. Wolf and Gibson (2006) do not claim that

anaphora is a (necessary) criterion for ELABORATION. They formulate more generally that "[o]ften when there is an anaphoric relation between two discourse segments, these discourse segments are also related by a coherence relation" (p. 35).

ELABORATION relations have also been compared to focus structures (Knott et al., 2001) such as described in Centering Theory (Grosz and Sidner, 1986; Grosz et al., 1995), which models anaphora across adjacent sentences.

Though in none of the definitions cited above it is explicitly said that in an ELABORATION relation between two discourse segments, a discourse entity or referent in N is continued in S by a co-specified linguistic expression, it is the case in many examples that we found in the literature including those presented above. Terms like "situation", "element of subject matter", "subject", "entity", and "event" seem to refer to different types of discourse entities.

Because of this frequent association of ELABORATION with semantic relations between certain distinguished discourse entities, we also believe that it can be compared to types of "thematic progression" or "thematic development" known from text linguistics. The following is a simplified description of types of thematic progression as introduced in Daneš (1970) and Zifonun et al. (1997):

1. Continuation of theme or rheme
2. Derivation or integration from the preceding theme or hypertheme
    a) derivation from hypertheme
    b) derivation from preceding theme or rheme
    c) integration of preceding themes in one hypertheme

Thematic relations between segments with a common topic abound in any given text, and according to (Carlson et al., 2001, p. 53) ELABORATION is "extremely common at all levels of the discourse structure" as well. In our corpus, it is the most frequent relation (43% of all relations in the SemDok-corpus and 38% of all relations in the subcorpus used for the analyses described in this article, see Section 3.1). ELABORATION is much less constrained than most other RST relations and seems to be a natural "default relation" to be assigned when no other relation can be assigned due to an absence of lexical discourse markers (another candidate for a default relation is LIST).

In order to render the original RST-definition of ELABORATION by (Mann and Thompson, 1988, p. 273) more detailed, we extended the set of rhetorical relations for our annotation project with subtypes of ELABORATION and with definitions which make reference to discourse entities and themes. In doing so, we also compared other sets of subtypes of ELABORATION found in the research literature, i.e. Mann and Thompson (1988), Hovy and Maier (1995), and Carlson et al. (2001), with relation instances in our corpus labelled as ELABORATION. The hierarchy of ELABORATION relations used in the final version of our corpus annotation scheme is shown in Figure 2. In the annotation of the sample corpus described in Section 3.3 , annotators where asked to use only the terminal types, i.e. the leaves of the hierarchy for annotation, except for those types that are marked with an asterisk '*' in Figure 2. Only if annotators definitely could

not decide on one terminal type were they allowed to annotate one of the intermediate types.

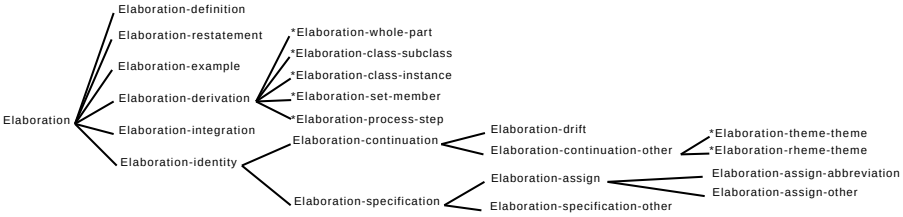


Figure 2: SemDok hierarchy of ELABORATION relations

The first three subrelations ELABORATION-DEFINITION, ELABORATION-RESTATEMENT, and ELABORATION-EXAMPLE are not defined in terms of thematic progession or referential continuation, but rather along the lines of the relations DEFINITION, EXAMPLE, and RESTATEMENT in Carlson et al. (2001), and in the annotation task, they take priority over an assignment of one of the remaining subtypes. ELABORATION-DEFINITION holds when the satellite contains a definition of a technical concept occurring in the nucleus. In our corpus, it is frequently signalled by a colon terminating the nucleus, and/or XML markup such as the DocBook <`glossentry`> element on the annotation layer of logical document structure (cf. Walsh and Muellner, 1999). ELABORATION-EXAMPLE holds, when the satellite represents an example of the nucleus or of a concept in the nucleus. It is generally accompanied by a lexical discourse marker in the satellite such as *z.B.* or *beispielsweise* (cf. Example (7)). Finally, ELABORATION-RESTATEMENT holds, when the satellite represents a reformulation of the nucleus of about the same length.

The subrelation ELABORATION-IDENTITY, on the other hand, is characterised by a thematic or referential identity between nucleus and satellite. In case of its subtype ELABORATION-CONTINUATION, there is thematic continuity between nucleus and satellite either in the form of a common hypertheme (subtype ELABORATION-DRIFT) or in the form of an explicit linguistic expression in the satellite that refers to the rheme or theme of the nucleus (subtype ELABORATION-CONTINUATION-OTHER[4]). ELABORATION-DRIFT is further defined to cover the following cases: a.) The hypertheme need not necessarily be mentioned in the nucleus or satellite, but it should be nameable, b.) a theme that was introduced in the nucleus as an NP is continued in the satellite in an embedded phrase only (cf. Example (11)), or c.) a thematic event anaphor (like *dies*) in the satellite refers to the proposition or set of propositions that forms the nucleus (cf. Example (10)).

In case of the other subtype of ELABORATION-IDENTITY, ELABORATION-SPECIFICATION, the satellite is about the same discourse entity in such a way that the meaning of

[4]The suffix "-other" was used to distinguish the major subtype of ELABORATION-CONTINUATION, ELABORATION-SPECIFICATION, and ELABORATION-ASSIGN from its co-subtypes, respectively.

the nucleus is extended, restricted or further specified by a modifying phrase only, i.e. as an incomplete sentence and without explicitly mentioning the thematic discourse entity again. Its subtype ELABORATION-ASSIGN holds, when the meaning of the nucleus is in a way *assigned* by the author to the expression in the satellite. In academic texts, this frequently occurs when abbreviations or acronyms are introduced (subtype ELABORATION-ASSIGN-ABBREV). ELABORATION-ASSIGN is thus similar to ELABORATION-DEFINITION, but with inverse nuclearity. The regular instances of ELABORATION-SPECIFICATION which are not covered by ELABORATION-ASSIGN, are labelled ELABORATION-SPECIFICATON-OTHER (see Example (9)).

(7) ELABORATION-EXAMPLE:[5] [*Åland hat auch in vielen anderen Hinsichten eigene Gesetze,*]$_{Nuc}$ [*z.B. sind die Inseln entmilitarisiert.*]$_{Sat}$

(8) ELABORATION-CONTINUATION-OTHER:[6] [*Im folgenden Abschnitt werden wir zunächst einige terminologische Klärungen vornehmen.*]$_{Nuc}$ [*Diese betreffen einerseits unser Verständnis von regionalen Varietäten (2.1), andererseits das Spracheinstellungskonzept (2.2).*]$_{Sat}$

(9) ELABORATION-SPECIFICATION-OTHER:[7] [*Ob regionale Varietäten* [*(Dialekte, Regionalsprachen, nationale Standardvarietäten)*]$_{Sat}$ *Thema des Deutsch als Fremdsprache-Unterrichts sein können bzw. sein sollten, ist in den letzten Jahren zunehmend zum Gegenstand kontroverser Diskussionen geworden.*]$_{Nuc}$

(10) ELABORATION-DRIFT:[8] [*Die vorherrschende Meinung insbesondere bei DaF-Lehrern und bei den meisten Lehrbuchverlagen scheint zu sein, dass sich der DaF-Unterricht hauptsächlich auf die Vermittlung der deutschen Standardsprache beschränken muss und soll.*]$_{Nuc}$. [*Dies spiegelt sich zum einen in der Vernachlässigung regionaler Varietäten in DaF-Lehrwerken zugunsten der Standardsprache wider {...}*]$_{Sat}$

(11) ELABORATION-DRIFT:[9] [*Automatisierte Prozesse im L2-Erwerb sind solche, auf die keine oder nur geringe Aufmerksamkeit gerichtet wird.*]$_{Nuc}$ [*Eine wichtige Funktion der Automatisierung ist die Freisetzung von Kapazitäten für die gleichzeitige Bewältigung von aufmerksamkeitsintensiven Aktivitäten.*]$_{Sat}$

(12) ELABORATION-DERIVATION:[10] [*Die Erhebung und Analyse der mündlichen Primärdaten erfolgt in zwei großen Blöcken.*]$_{Nuc}$ [*In einer Querschnittsuntersuchung wird zunächst die Frage untersucht, wie {...}. Hiervon ausgehend können im zweiten Block longitudinal Veränderung von {...} verfolgt werden.*]$_{Sat}$

ELABORATION-DERIVATION, which is another direct subtype of ELABORATION, is

[5] Example taken from Mirja Saari (2000): "Schwedisch als die zweite Nationalsprache Finnlands: Soziolinguistische Aspekte". In: *Linguistik Online 7*, http://www.linguistik-online.de.

[6] Example taken from Harald Baßler, Helmut Spiekermann (2001): "Dialekt und Standardsprache im DaF-Unterricht. Wie Schüler urteilen - wie Lehrer urteilen". In: *Linguistik Online 9*, http://www.linguistik-online.de.

[7] Example taken from Baßler/Spiekermann (2001).

[8] Example taken from Baßler/Spiekermann (2001).

[9] Example taken from Olaf Bärenfänger, Sabine Beyer (2001): "Zur Funktion der mündlichen L2-Produktion und zu den damit verbundene kognitiven Prozessesn für den Erwerb der fremdsprachlichen Sprechfertigkeit". In: *Linguistik Online 8*, http://www.linguistik-online.de.

[10] Example taken from Bärenfänger/Beyer (2001).

based on thematic derivation, i.e. comprises whole-part, class-subclass, class-instance, set-member, or process-step relations between entities in the nucleus and the satellite (cf. Example (12)). ELABORATION-INTEGRATION is its opposite, with the inverse relation pairs, i.e. part-whole, subclass-class etc.

Only few of our subrelations are accompanied by explicit lexical, grammatical, or punctuational discourse markers, e.g. ELABORATION-EXAMPLE (*z.B.*) or ELABORATION-SPECIFICATION (parenthesis and phrase status of satellites), but the most frequently occurring subtypes of ELABORATION are not signalled by explicit discourse markers and cannot automatically be determined on the basis of lexical or grammatical cues.

2.3 Research questions and hypotheses

Based on our understanding of ELABORATION as indicating thematic relations in the framework of RST, it seems reasonable to look for the cues that are also used for the analysis of thematic relations. One prominent signal of thematic connections are referential ties between adjacent sentences, or more specifically: references between sentence themes (cf. Daneš, 1970; Givón, 1983, 1992). Sentence themes are signalled by nominal discourse entities, often expressed as pronouns, definite NPs, NPs in sentence initial position, or NPs in the role of grammatical subject. Anaphoric relations between adjacent discourse segments should therefore be good indicators for thematic relations, and hence for ELABORATION. Figures 3 and 4 exemplify this interrelationship: In the former figure, the cue for the discourse relation is a lexical discourse marker whereas in the latter figure, the discourse relation has an anaphoric relation as its cue.

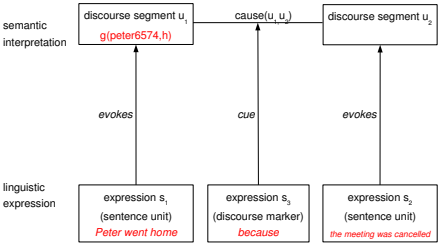
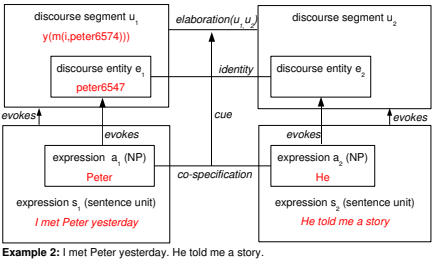


Figure 3: Linguistic expressions and semantic interpretation with lexical discourse marker

Generally, we expect that anaphora is a *necessary* condition for Elaboration while we also want to test whether it could be a *sufficient* condition. Furthermore, we expect that specific anaphoric relations from the scheme introduced in Section 2.1 correspond to specific ELABORATION relations; we would, for example, expect that ELABORATION-CONTINUATION-OTHER is indicated by the anaphoric relation *cospec:ident*. An overview of expected correspondences between thematic relations, ELABORATION relations and



Example 2: I met Peter yesterday. He told me a story.

Figure 4: Linguistic expressions and semantic interpretation with anaphora

anaphoric relations is given in Table 1. (Only those ELABORATION subrelations for which we had expectations are shown.).

Table 1: Theoretical correspondences between thematic relations, ELABORATION relations and anaphoric relations

Thematic Relations	**Elaboration Relations**	**Anaphoric Relations**
Continuation of theme or rheme	ELABORATION-CONTINUATION-OTHER	*cospec:synonym* *cospec:paraphrase* *cospec:ident*
Derivation from preceding theme or rheme	ELABORATION-DERIVATION	*cospec:hyperonym* *bridging:holonym* *bridging:setMember*
Integration of preceding theme or themes in one hypertheme	ELABORATION-INTEGRATION	*bridging:meronym* *cospec:hyponym* *bridging:hasMember*
Derivation from hypertheme	ELABORATION-DRIFT	*bridging:bridging* *bridging:poss* *bridging:abstrProp* *bridging:abstrCluster*

Since ELABORATION-CONTINUATION-OTHER should have been annotated when an explicit linguistic expression refers to the theme or rheme of the nucleus, we would expect it to be accompanied by semantic relations between discourse entities that indicate referential identity, i.e. *cospec:synonym*, *cospec:paraphrase*, or *cospec:ident*. Since ELABORATION-DERIVATION is based on *whole-part*, *class-subclass*, *class-instance* *set-member*, *process-step* (terms from the definition by Mann and Thompson, 1988) relations between entities in nucleus and satellite, we would expect it to be accompanied by the semantic relations *cospec:hyperonym*, *bridging:holonym*, or *bridging:setMember*. Since ELABORATION-INTEGRATION is the opposite we would expect it to appear together with *bridging:meronym*, *cospec:hyponym*, or*bridging:hasMember*. As *Elaboration-drift* may hold due to a common hypertheme, it may firstly appear together with *bridg-*

ing:bridging; since it may hold on account of thematic continuation realised in an embedded phrase, it may secondly be accompanied by *bridging:poss* (In an NP like *seine Untersuchung*, the possessive pronoun takes the position of an NP in genitive which is embedded in the whole, higher-level NP as a whole whose head is *Untersuchung*). Thirdly, since ELABORATION-DRIFT may hold when a thematic continuation is realised by an event anaphor, it may be accompanied by a *bridging:abstrProp* or *bridging:abstrCluster* relation.

Since non-thematic anaphoric relations between discourse segments might theoretically hold as well, one research question is whether the theoretical correspondences in Table 1 work as practical indicators of ELABORATION. Our general goal is to investigate inhowfar our theoretically derived claims are supported by empirical evidence by analysing a corpus that has been annoted on the level of anaphoric structure and on the level of rhetorical structure.

Our corpus, its relevant linguistic annotations and the analysis tools are described in the following section.

3 Methods

3.1 Corpus

The SemDok corpus used both for research on discourse structure and anaphoric structure consists of 47 German linguistic scientific journal articles, formally annotated on the levels of syntax, morphology and document structure. For the analysis of correlations between anaphoric relations and ELABORATION relations we developed a sample corpus, which comprises two scientific journal articles from the SemDok corpus, one web-published scientific article and one newspaper article (altogether 15.622 word forms). These four texts were segmented in elementary and complex discourse segments, and annotated on the levels of rhetorical structure (RST-HP, for RST, hypotaxis and parataxis) and discourse entities and anaphoric relations (CHS, for cohesion). The two kinds of annotations have been carried out independently.

3.2 Annotation of anaphoric structure

The corpus under investigation has been annotated manually for anaphoric relations using the annotation tool *Serengeti* which is described in detail together with the annotation scheme in Diewald et al. (2008, this volume). Anaphoric relations are marked between text spans, i.e. between linguistic units (*markables*). These text spans evoke discourse entities as part of the discourse universe, thus anaphoric relations are *marked* between linguistic units but the corresponding semantic relations *hold* between discourse entities.

Each text of the corpus has been preprocessed using the dependency parser Machinese Syntax[11] which provides lemmatisation, POS information, dependency structure, mor-

[11] `http://www.connexor.eu`.

phological information and grammatical function. Based on this information, markables of nominal type have been detected automatically by identifying nominal heads (i.e. nouns or pronouns) and their premodifiers.

The annotation procedure has been performed in two steps. The first step has been done before the data analysis and its focus lay on the annotation of anaphoric relations between nominal anaphors and antecedents of nominal type only. The second annotation step has been done after the analysis of the nominal data of step 1 and its focus lay on the annotation of abstract entity anaphora including adverb anaphors (cf. Figure 1).

During the first step, a complete annotation has been done for those anaphoric relations with both anaphor and antecedent of nominal type. These relations include pronominal anaphors as well as definite description anaphors with nominal antecedents where both intra- as well as inter-sentential anaphora has been taken into account.

In a second step, abstract entity anaphora has been annotated. These relations hold between the anaphor and an antecedent of propositional or event type. Whereas the first step has been a complete annotation of both markables and anaphoric relations the second step has been a partial annotation, only. Due to the vast amount of all propositions and events in a text, only those discourse entities have been identified as markables that form the antecedent of an anaphoric relation. Three types of discourse entities have been annotated manually: The type CLUSTER describes discourse entities that are evoked by several adjacent sentences, PROP describes entities evoked by one proposition (sentence or embedded clause), and EVTYPE describes all entities evoked by a verb and its arguments. Furthermore, all adverbial anaphors (such as *hierbei*, *dabei*) have been marked as discourse entity of type ADV in order to annotate adverb anaphora leading to a total number of five different types of discourse entities: NOMINAL, ADV, PROP, EVTYPE, and CLUSTER.

For the corpus under investigation a total number of 662 anaphoric relations has been annotated during the first step; during the second step another 68 abstract entity anaphors have been annotated.

3.3 Annotation of rhetorical structure

Rhetorical structure according to RST was encoded in the XML application RST-HP developed in the project SemDok (Lüngen et al., 2008). Discourse segments are marked using the two elements `hypo` and `para` with a relation name in the `@relname` attribute (see Lüngen et al., 2006, 2008, for a description and sample annotations of RST-HP). Unlike URML (Reitter and Stede, 2003) and the XML-like format put out by the RSTTool (O'Donnell, 2000), RST-HP exploits the XML document tree to represent an RST tree, which means that general XML query tools such as XPath or the Sekimo Tools (Witt et al., 2005) can be applied straightforwardly to query RST-HP annotations.

In manual or automatic annotation, rhetorical relations are assigned on the basis of the *RRSet*, a taxonomy comprising 70 rhetorical relation types for the analysis of the discourse structure of scientific articles, 44 of which are base types to be used in the manual and automatic annotations (Bärenfänger et al., 2008). The ELABORATION

sub-hierarchy given in Figure 2 is part of the RRSet taxonymy. The annotation guidelines stated that when a lexical discourse marker for an ordinary relation could be found, this relation should be annotated while the conditions for ELABORATION need not be checked. This procedure, which as we think is typical for RST analyses, gives ELABORATION the status of a default relation.

The RST annotation of the four articles of the sample corpus was done using O'Donnell's RSTTool. The XML-like format that is output by the RSTTool was converted to RST-HP by means of a perl program. Each file was annotated independently by two annotators, who then discussed possible annotation differences and agreed on a single "master" version which was subsequently used in the comparison with annotations of anaphoric structure described below.

For the present study, we concentrated on subtrees of RST trees for complex discourse segments of type "block", i.e. trees where the minimal units are *elementary discourse segments* (basically clause-like units) and whose root node corresponds to a paragraph. The RST-HP annotations for block segments constructed in the sample corpus contained 846 RST subtrees altogether.

To get an idea of inter-annotator agreement for the RST relation assignment task, we measured agreement within "block" segments for three articles that were coded by three annotators each. Kappa values for the nine resulting annotator pairings ranged between 0.47 and 0.81 which is interpreted as 'moderate agreement' to 'almost perfect agreement' by Landis and Koch (1977).

3.4 Analysis

During the annotation of anaphoric and rhetorical structure, the primary data of the input documents were left unchanged so that the Sekimo query tools could be employed for querying relations between elements of two XML annotation layers (cf. Witt et al., 2005). We focused on the analysis of the *inclusion* relation to verify whether a discourse entity on the CHS layer was included in a discourse segment on the RST-HP annotation layer.

In order to research the hypotheses formulated in Section 2.3, we firstly derived the set of instances of adjacent discourse segments DS_i and DS_j that contained an anaphoric expression in DS_j whose antecendent was contained in DS_i, together with the information of whether DS_i and DS_j formed a combined RST subtree in RST-HP with a relation assignment or not. This query resulted in an XML dataset of 662 anaphoric instances. Secondly, we derived the set of instances of adjacent discourse segments that formed a combined RST subtree in RST-HP, together with the information about an occurrence of anaphora formed by an anaphoric expression in DS_j and a related antecedent in DS_i, and if applicable, its type. This query resulted in an XML dataset of 846 relation instances. To obtain the statistics reported in Section 4, these two databases were queried using XPath expressions.

4 Results and Discussion

4.1 Is anaphora a sufficient condition for ELABORATION?

That the existence of an anaphoric relation might not be a sufficient condition for the discourse relation of ELABORATION to hold seems obvious as anaphora can also be involved in other relations. Most other relations are defined without recourse to referential structure or thematic progression, and are frequently signalled by a lexial discourse marker. But in order to quantify the degree in which anaphora might or might not be a sufficient condition for ELABORATION, we checked all anaphoric instances for their co-occurrence with ELABORATION. The results of this investigation are given in Table 2.

Table 2: Is anaphora a sufficient condition?

	Total No.
anaphoricInstance	662
@rtype='elaboration'	176
@rtype='no-RST-relation'	301
@rtype='RST-relation-other-than-elaboration'	185

Due to the fact that ELABORATION is a default relation, we had expected anaphoric relations to coincide with relations other than ELABORATION: 185 out of 662 anaphoric instances (27,95%) coincide with relations other than ELABORATION whereas 176 anaphoric instances (26.59%) coincide with ELABORATION.

Interestingly, the majority of anaphoric instances (45,47%) does not coincide with any relation at all. These instances are either located within the same discourse segment or there is no rhetorical relation between the relevant segments due to the overall discourse segmentation.

Clearly, the occurrence of an anaphoric relation is not a sufficient condition for ELABORATION. In the following section we will investigate the question whether the existence of an anaphoric relation is a *necessary condition* for ELABORATION.

4.2 Is anaphora a necessary condition for ELABORATION?

In the corpus, 298 ELABORATION instances could be identified on the basis of the first annotation step, but for only 191 of them, an anaphoric relation holds between discourse entities in the related discourse segments. In 107 cases, ELABORATION does not correlate with an anaphoric relation (Table 3). The different subtypes of ELABORATION deviate with respect to the strength of their interrelation with anaphoric relations. ELABORATION-CONTINUATION-OTHER correlates almost always with an anaphoric relation, whereas ELABORATION-SPECIFICATION-OTHER and ELABORATION-ASSIGN-OTHER are only weakly associated with anaphoric relations – for them, there are more occurrences without an anaphoric relation present than with an anaphoric relation. How can these differences be explained?

Firstly, there is the technical reason that in the definitions of ELABORATION-SPECIFICATION-OTHER and ELABORATION-ASSIGN-OTHER, the satellite is described to have phrasal status (i.e. not clausal), and such units mostly correspond to parenthetical segments. Anaphoric relations to discourse entitities in parentheses, however, were not marked on the CHS annotation layer.

Secondly, neither ELABORATON-SPECIFICATION-OTHER, ELABORATION-ASSIGN-OTHER, ELABORATION-DEFINITION, nor ELABORATION-EXAMPLE are typical thematic continuations or derivations. Instead of being signalled by referential ties, they are indicated by lexical and syntactic cues (cf. Section 2.2): ELABORATION-EXAMPLE is almost always marked by lexical markers like "z.B." or "beispielsweise", and ELABORATION-SPECIFICATION-OTHER and ELABORATION-ASSIGN-OTHER are indicated by parentheses or brackets which encloses the NPs or PPs in the satellite that specifies, extends or restricts an entity in the nucleus without repeating the entity itself.

Another relation which shows a different behaviour than expected is ELABORATION-DRIFT. Although this relation is defined as exhibiting some sort of thematic continuity, it does not – like ELABORATION-CONTINUATION-OTHER – frequently correlate with anaphoric relations (see Table 3). 44 out of 111 instances of ELABORATION-DRIFT are not connected by an anaphoric relation at all. These result was so much against our expectations that we decided to carry out a qualitative analysis of the 107 ELABORATION instances which had no correspondence with an anaphoric relation.

Table 3: Number of ELABORATION instances with anaphoric relations

	All	**With anaphoric relations**		**Without anaphoric relations**	
ELABORATION-DRIFT	111	67	(60.36%)	44	(39.64%)
ELABORATION-CONTINUATION-OTHER	56	51	(91.07%)	5	(8.93%)
ELABORATION-SPECIFICATION-OTHER	43	20	(46,51%)	23	(53,49%)
ELABORATION-DERIVATION	36	28	(77.78%)	8	(22.22%)
ELABORATION-DEFINITION	13	6	(46,15%)	7	(53,85%)
ELABORATION-EXAMPLE	10	4	(40%)	6	(60%)
ELABORATION-INTEGRATION	7	4	(57,14%)	3	(42,86%)
ELABORATION-IDENTITY	7	4	(57,14%)	3	(42,86%)
ELABORATION-ASSIGN-OTHER	6	1	(16,67%)	5	(83,33%)
ELABORATION	5	4	(80.00%)	1	(20.00%)
ELABORATION-RESTATEMENT	3	1	(33,33%)	2	(66,67%)
ELABORATION-CONTINUATION	1	1	(100.00%)	0	(0.00%)
All Elaboration-Trees	298	191	(64,09%)	107	(35,91%)

The qualitative analysis showed that a bulk of the missing anaphoric relations were due to the scope of the anaphoric relation set and the annotation focus chosen in the project Sekimo, which was on nominal antecedents only. Propositional antecedents had not been taken into account during the first annotation phase. In 37 of the 107 not anaphorically linked ELABORATION instances, anaphoric relations could – according to the findings of the qualitative analysis – be established on the basis of a propositional

antecedent. These abstract entity anaphors were then annotated in a second annotation step.

For another 38 instances it was possible to assign types of anaphoric relations that are not based on lexical-semantic relations, but involved other, e.g. morpho-semantic relations (e.g. derivation) or broad association (such as *Kind – Infantilisierung* in the sample corpus). Whereas anaphora due to identity of head nouns or due to lexical-semantic relations can be decided rather unambiguously, this is not the case for anaphora based on association. Narrow association (e.g. *wedding – bride*) is detected more easily than broad association. But taking broad association into account helped to identify additional anaphoric relation instances such that subsequently only six instances (i.e. 2,69%) of the 223 instances related by ELABORATION-CONTINUATION-OTHER, ELABORATION-DRIFT, ELABORATION-DERIVATION, ELABORATION-INTEGRATION, and ELABORATION-DEFINITION had no anaphoric connection. Table 4 shows the effect of the qualitative analysis as well as of the second annotation step.

Table 4: ELABORATION instances with anaphoric relations after qualitative analysis and second annotation step

	All	**With anaphoric relations**		**Without anaphoric relations**	
ELABORATION-DRIFT	111	108	(97.30%)	3	(2.70%)
ELABORATION-CONTINUATION-OTHER	56	55	(98.21%)	1	(1.79%)
ELABORATION-SPECIFICATION-OTHER	43	25	(58.14%)	18	(41.86%)
ELABORATION-DERIVATION	36	35	(97.22%)	1	(2.78%)
ELABORATION-DEFINITION	13	12	(92.31%)	1	(7.69%)
ELABORATION-EXAMPLE	10	7	(70.00%)	3	(30.00%)
ELABORATION-INTEGRATION	7	7	(100%)	0	(0%)
ELABORATION-IDENTITY	7	7	(100%)	0	(0%)
ELABORATION-ASSIGN-OTHER	6	1	(16.67%)	5	(83.33%)
ELABORATION	5	5	(100%)	0	(0%)
ELABORATION-RESTATEMENT	3	3	(100%)	0	(0%)
ELABORATION-CONTINUATION	1	1	(100%)	0	(0%)
All Elaboration-Trees	298	266	(89.26%)	32	(10.74%)

Altogether, the revised quantitative analysis of the correlations between ELABORATION and anaphoric relations shows that 108 of 111 instances of ELABORATION-DRIFT, 35 out of 36 instances of ELABORATION-DERIVATION, seven out of seven instances of ELABORATION-INTEGRATION and 55 out of 56 instances of ELABORATION-CONTINUATION-OTHER indeed co-occur with an anaphoric relation. Only the figures for ELABORATION-SPECIFICATION-OTHER, ELABORATION-ASSIGN-OTHER and ELABORATION-EXAMPLE did not differ significantly after the qualitative analysis. Our second hypothesis – that an anaphoric relation is a necessary condition for ELABORATION – must therefore be considered true for all subtypes of ELABORATION except ELABORATION-SPECIFICATION-OTHER, ELABORATION-ASSIGN-OTHER and ELABORATION-EXAMPLE. Note that the latter three relations comprise the majority of cases where ELABORATION is marked by a lexical discourse marker or by parenthesis.

In Table 1 we pointed out that specific subtypes of ELABORATION are expected to correspond to specific thematic relations and anaphoric relations. The hypothesised correspondences could be partly supported by the quantitative analysis of the corpus. The results differed with respect to their relative frequency. Stronger correlations with certain anaphora types were found for ELABORATION-CONTINUATION-OTHER, ELABORATION-DERIVATION and ELABORATION-INTEGRATION. The most frequent anaphoric relations contained after the first annotation step are shown in Table 5.[12]

Table 5: Co-occurrences of ELABORATION relations and anaphoric relations

Elaboration Instances	**With Anaphoric Relations Contained**
ELABORATION-CONTINUATION-OTHER 56 (51 with anaphora)	38 (63) x *cospec:ident* 6 (15) x *bridging:setMember* 6 (8) x *cospec:paraphrase* 6 (7) x *bridging:bridging* 3 (5) x *bridging:poss* 3 (3) x *cospec:synonym*
ELABORATION-DERIVATION 36 (28 with anaphora)	17 (43) x *bridging:setMember* 13 (17) x *cospec:ident* 3 (5) x *bridging:bridging* 3 (5) x *cospec:isA* 2 (4) x *cospec:synonym*
ELABORATION-INTEGRATION 7 (4 with anaphora)	2 (3) x *bridging:hasMember* 2 (2) x *cospec:paraphrase*
ELABORATION-DRIFT 111 (67 with anaphora)	41 (60) x *cospec:ident* 12 (13) x *bridging:bridging* 11 (13) x *cospec:paraphrase* 10 (17) x *bridging:setMember*

ELABORATION-CONTINUATION-OTHER co-occurs with *cospec:ident* most of the time (38 of 56 cases, i.e. 67.86% of all instances of ELABORATION-CONTINUATION-OTHER co-occur with *cospec:ident*), six co-occur with *cospec:paraphrase*. 17 of 36 instances of ELABORATION-DERIVATION co-occurred with *bridging:setMember*, and two of three instances of ELABORATION-INTEGRATION co-occurred with *bridging:hasMember*. By contrast, the findings for ELABORATION-DRIFT were much more ambiguous: It co-occurs with *cospec:ident* (41 of 111 instances), *cospec:paraphrase* (eleven of 111 instances), *bridging:bridging* (twelve of 111 instances) and *bridging:setMember* (ten of 111 instances). Despite these ambiguities, some types of anaphoric relations might help automatically identify a specific ELABORATION relation when no other rhetorical relations can be determined, and we report a test of this in Section 5.

The qualitative analysis of the corpus also suggested that anaphoric expressions that correlated with ELABORATION are more frequently found in sentence-initial position

[12] In the column entitled 'with anaphoric relations contained', the first figure represents the number of ELABORATION instances that contain at least one anaphoric instance of the type, and the second figure in brackets represents the total number of anaphoric instances of the type contained

(*vorfeld*) or in the role of the grammatical subject (e.g. in 42 of the 51 ELABORATION-CONTINUATION-OTHER instances with anaphora) than in a different position or role. This is presumably due to the fact that subject and vorfeld positions are typical *topic* (i.e. sentence theme) positions in German syntax.

5 Discourse parsing experiments

In order to evaluate the contribution of an analysis of anaphora to automated discourse parsing, we integrated a processing of anaphoric cues from the CHS annotation layer of an input document in the RST-based discourse parser developed in the SemDok project.

The central component of the parsing system is called GAP – Generalised Annotation Parser. GAP is a bottom-up passive chart parser implemented in Prolog. GAP is applied in a cascade architecture first to *elementary discourse segments* ("clause-like units"), second to *sentential discourse segments*, and third and further to different types of *complex discourse segments* ("block", "division", "document") specified on the initial discourse segment annotation layer. Each of these segment levels is provided with its own set of *reduce rules*. Reduce rules are binary rules that describe the conditions under which two adjacent discourse segments form a new (larger) discourse segment. They are mostly derived from a discourse marker lexicon that contains combinatorial information about conjunctions and discourse adverbs (cf. Lüngen et al., 2008).

The rule component for the sentential level (where input segments are sentential discourse segments, and the top nodes of complete RST trees correspond to paragraphs of the text) was altered in six different experiments. It originally contained 73 rules derived from (the readings of) lexical discourse markers such as *beispielsweise* (indicating ELABORATION-EXAMPLE), *aber* (indicating CONTRAST), or *danach* (indicating SEQUENCE).

According to the findings discussed in Section 4, we added three rules that make reference to the CHS annotation layer in the rule component (cf. Table 6).

Table 6: Reduce rules operating on annotation layer CHS. General condition for R_0, R_1, and R_2: DS_1 and DS_2 are two adjacent discourse segments without a lexical discourse marker pointing to a relation other than subtypes of ELABORATION, and A_2 is an anaphor in the first sentence of DS_2, and A_1 is its antecedent in DS_1.

Rule	**Reduce target**	**Constraints by type of link between A_1 and A_2 on CHS**
R0	N-S, ELABORATION-DRIFT	(no further constraints)
R1	N-S, ELABORATION-CONTINUATION-OTHER	*cospec:ident* OR *cospec:paraphrase* OR *cospec:synonym* OR *cospec:addInfo*
R2	N-S, ELABORATION-DERIVATION	*bridging:setMember* OR *bridging:meronym* OR *bridging:poss*

We also introduced a ranking of rule groups and implemented the strategy that adjacent discourse segment pairs are only to be tested against reduce rules of a higher rank when no rules of a lower rank have matched before. The rule groups and their ranks are:

1. Rules based on lexical discourse markers
2. Rules based on anaphora (newly introduced)
3. Default rule (reduce target is LIST-COORDINATION, or alternatively, ELABORATION-DRIFT)

Thus, an analysis of anaphora is only activated if no discourse marker indicating a rhetorical relation other than ELABORATION and its subtypes could be found on other annotation levels.

Based on the combinations of the two versions of the default rule and the rules R_0, R_1, R_2, we conducted several parsing experiments with an article from our corpus and with the different rule sets included in GAP. The article was one that was also in the subcorpus used for deriving the statistics, as at the time of the experiments, no other articles with an annotation of anaphora was available.

Experiment I comprised the rule set of the original parser with ELABORATION-DRIFT (the most frequently occurring subtype of ELABORATION in the sample corpus) as default relation and served as a baseline. In experiment II, we tested the original rule set with LIST-COORDINATION as default relation plus the assignment of ELABORATION-DRIFT whenever any kind of anaphoric relation was found between two discourse entities in DS_1 and DS_2 (R_0 in Table 6). Experiment III comprised the original rule sets and rules R_1 and R_2 with conditions derived from the corpus study for the assignment of ELABORATION-CONTINUATION-OTHER and ELABORATION-DERIVATION according to the type of anaphora (cf. Table 6). The performance results for these discourse parsing experiments are shown in Table 7.

For deriving the figures in the column entitled "RRSet 30", the `relname` attribute in the reduce rules and in the master annotations were re-labelled by mapping all instances of subtypes of ELABORATION on one generic ELABORATION label.

Table 7: Results for discourse parsing experiments with and without anaphora processing

	Anaphora processing	**Default Relation**	**RRSet 44**		**RRSet 44**	**RRSet 30**
			Prec	**Rec**	**Rec max**	**Rec max**
I	No (Baseline)	ELABORATION-DRIFT	34.06	34.83	38.20	42.70
II	Rule R_0	LIST-COORDINATION	35.16	35.96	38.20	43.82
III	Rules R_1, R_2	LIST-COORDINATION	37.36	38.20	41.58	44.94

Using the full RRSet with 44 categories, the parser in experiment III, which included rules about subtypes of ELABORATION relations derived from specific types of anaphoric relations, performed best with a recall of 38.20% (precision 37.36%). The general assignment of the most frequent subrelation ELABORATION-DRIFT in case of an occurrence of

any kind of anaphora between DS_1 and DS_2 (experiment II) performed worse than the baseline. In experiments II and III, precision was also improved in comparison with the baseline, because rules R_1 and R_2 are more specific than the default rule of the baseline and thus filter out more hypotheses. In the third column entitled "Rec max", the maximum recall, i.e. the recall that can be reached when the whole, unpruned chart is matched against the reference file, is shown.

In the fourth column, the maximum recall for a praser with reduced relation set of 30 categories, where all subtypes of ELABORATION are represented by the general ELABORATION label is shown. The four series of experiments represented by each column all show the tendency that the performance gets better when constraints about anaphora are added (in the Rec max experiments the precision lay between 12 and 19% and showed the same tendency). However, since the increases of percentages rely on a handful of relation labels only, experiments with more documents are needed to confirm this.

6 Conclusions

Anaphoric (coreference) structure and relational, hierarchical discourse structure are two aspects of the description of coherence in discourse. In several theories of relational discourse structure, anaphora, i.e. semantic relations between discourse entities play a role in defining the ELABORATION relation. Semantic relations between (topical) discourse entities are also the basis of text structures described by thematic progression analyses. Hence we refined the original definition of ELABORATION by introducing subtypes according to different types of thematic development. In discourse analyses in the form of RST annotation of text, the ELABORATION relation was assigned to two adjacent discourse segments when no discourse markers for other standard relations like CONTRAST or SEQUENCE are available. Furthermore, we introduced a framework for the annotation of anaphora.

For an empirical investigation of the relation between discourse anaphora and discourse structure we statistically analysed a corpus that was independently annotated on the levels of anaphoric structure and rhetorical structure. The focus of the investigation has been on ELABORATION relations and whether anaphora can serve as a cue for ELABORATION, because unlike other RST relations, most subtypes of ELABORATION lack associations with lexical discourse markers. The research questions guiding our analyses were whether anaphora could be used as a necessary and/or sufficient criterion for ELABORATION, whether subtypes of ELABORATION correlate with specific subtypes of anaphora, and whether anaphora could be used as a cue in automated discourse analysis.

According to our results, anaphora is not a sufficient condition for ELABORATION, i.e. a large percentage of anaphoric instances was connected to relations other than ELABORATION. Still, anaphora seems to be a necessary condition for most subtypes of ELABORATION. The latter finding could be established after additionally annotating abstract entity anaphora in the corpus, which is frequently correlated with the subtype of

ELABORATION-DRIFT. Four ELABORATION subtypes were fairly ambiguous with respect to correlated anaphora types, but particularly ELABORATION-CONTINUATION-OTHER, ELABORATION-DERIVATION, and ELABORATION-INTEGRATION were strongly associated with *cospec:ident*, *bridging:setMember*, and *bridging:hasMember*, respectively.

The results of six discourse parsing experiments with one journal article, introducing rules operating on the CHS annotation layer in the discourse parser developed in the SemDok project, do suggest that a detailed analysis of anaphora types may help identify instances of specific subtypes of ELABORATION relations better, although the results of the test runs with a more informed evaluation of anaphora were only slightly better than those where ELABORATION was always assigned as a default relation when no other discourse marker was present.

The fact that anaphora is not a sufficient condition for ELABORATION, and the fact that ELABORATION is frequently used as a default relation could also be taken as arguments for introducing a *thematic level* as a separate and self-contained level of discourse analysis and annotation that complements RST analyses as suggested in Stede (2007). But then in order not to introduce redundancy into the representation of discourse, we think that one would also have to remove ELABORATION from the RST relation set and to relax the connectedness constraint of Mann and Thompson (1988).

References

Asher, N. (1993). *Reference to abstract objects in discourse*, volume 50 of *Studies in Linguistics and Philosophy*. Kluwer Academic Publishers, Dordrecht.

Asher, N. and Lascarides, A. (2003). *Logics of Conversation*. Cambridge University Press, Cambridge, UK.

Bärenfänger, M., Lobin, H., Lüngen, H., and Hilbert, M. (2008). OWL ontologies as a resource for discourse parsing. *LDV-Forum. GLDV-Journal for Computational Linguistics and Language Technology*, 23(2):17–26.

Carlson, L. and Marcu, D. (2001). Discourse tagging reference manual. Technical report, Information Science Institute, Marina del Rey, CA. ISI-TR-545.

Carlson, L., Marcu, D., and Okurowski, M. E. (2001). Building a discourse-tagged corpus in the framework of rhetorical structure theory. In *Proceedings of the 2nd SIGDIAL Workshop on Discourse and Dialogue*, Eurospeech 2001, Denmark.

Clark, H. (1977). Bridging. In Johnson-Laird, P.N. & Wason, P., editor, *Thinking: Readings in Cognitive Science*, pages 411–420. Cambridge University Press, Cambridge.

Corston-Oliver, S. (1998). *Computing of Representations of the Structure of Written Discourse*. PhD thesis, University of California, Santa Barbara.

Cristea, D., Ide, N., Marcu, D., and Tablan, M.-V. (2000). Discourse structure and co-reference: An empirical study. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING'2000)*, Luxembourg.

Cristea, D., Ide, N., and Romary, L. (1998). Veins theory: A model of global discourse cohesion and coherence. In *Proceedings of ACL/COLING'98*, pages 281–285, Montreal.

Daneš, F. (1970). Zur linguistischen Analyse der Textstruktur. *Folia Linguistica*, 4:72–78.

Diewald, N., Stührenberg, M., Garbar, A., and Goecke, D. (2008). Serengeti – webbasierte annotation semantischer relationen. *LDV-Forum. GLDV-Journal for Computational Linguistics and language Technology*.

Givón, T. (1983). Topic continuity in discourse: An introduction. In Givón, T., editor, *Topic Continuity in Discourse: A Quantitative Cross-Language Study*, pages 5–41. John Benjamins, Amsterdam, Philadelphia.

Givón, T. (1992). The grammar of referential coherence as mental processing instructions. *Linguistics*, 30:5–55.

Goecke, D., Stührenberg, M., and Holler, A. (2007). Koreferenz, Kospezifikation und Bridging: Annotationsschema. Interne Reports der DFG-Forschergruppe 437 "Texttechnologische Informationsmodellierung".

Goecke, D., Stührenberg, M., and Witt, A. (2008). Influence of Text Type and Text Length on Anaphoric Annotation. In (ELRA), E. L. R. A., editor, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco.

Grosz, B. J., Joshi, A. K., and Weinstein, S. (1995). Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21:203–225.

Grosz, B. J. and Sidner, C. L. (1986). Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Harman, D. and Liberman, M. (1993). TIPSTER complete. Philadelphia: Linguistic Data Consortium.

Holler, A., Maas, J.-F., and Storrer, A. (2004). Exploiting coreference annotations for text--to-hypertext conversion. In *Proceedings of the 4th International Conference on Language Resources and evaluation (LREC 2004)*, volume II, pages 651–654, Lissabon.

Holler-Feldhaus, A. (2004). Koreferenz in Hypertexten: Anforderungen an die Annotation. *Osnabrücker Beiträge zur Sprachtheorie (OBST)*, pages 9–29.

Hovy, E. and Maier, E. (1995). Parsimonious or profligate: How many and which discourse structure relations? Unpublished paper, `http://www.isi.edu/natural-language/people/hovy/publications.html`.

Kamp, H. and Reyle, U. (1993). *From Discourse to Logic*. Kluwer, Dordrecht.

Karttunen, L. (1976). Discourse referents. *Syntax and Semantics: Notes from the Linguistic Underground*, 7:363–385.

Knott, A., Oberlander, J., O'Donnell, M., and Mellish, C. (2001). Beyond elaboration: The interaction of relations and focus in coherent text. In Sanders, T., Schilperoord, J., and Spooren, W., editors, *Text representation: Linguistic and psycholinguistic aspects*, volume 8 of *Human Cognitive Processing*, pages 181–196. Benjamins, Amsterdam.

Landis, J. and Koch, G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33:159–174.

Le Thanh, H., Abeysinghe, G., and Huyck, C. (2004). Generating discourse structures for written texts. In *Proceedings of COLING'04*, Geneva, Switzerland.

Lüngen, H., Bärenfänger, M., Hilbert, M., Lobin, H., and Puskàs, C. (2008). Discourse relations and document structure. In Metzing, D. and Witt, A., editors, *Linguistic modeling of information and Markup Languages. Contributions to language technology*, Text, Speech and Language Technology. Springer, Dordrecht. To appear.

Lüngen, H., Lobin, H., Bärenfänger, M., Hilbert, M., and Puskàs, C. (2006). Text parsing of a complex genre. In *Proceedings of the Conference on Electronic Publishing (ELPUB)*, pages 247–256, Bansko, Bulgaria.

Mann, W. C. and Thompson, S. A. (1988). Rhetorical Structure Theory: Toward a functional theory of text organisation. *Text*, 8(3):243–281.

Marcu, D. (2000). *The Theory and Practice of Discourse Parsing and Summarization*. MIT Press, Cambridge, MA.

O'Donnell, M. (2000). RSTTool 2.4 – A markup tool for Rhetorical Structure Theory. In *Proceedings of the International Natural Language Generation Conference (INLG'2000)*, pages 253 – 256, Mitzpe Ramon, Israel.

Polanyi, L. (1988). A formal model of discourse structure. *Journal of Pragmatics*, 12:601–638.

Polanyi, L., van den Berg, M., and Ahn, D. (2003). Discourse structure and sentential information structure. *Journal of Logic, Language and Information*, 12:337–350.

Reitter, D. and Stede, M. (2003). Step by step: Underspecified markup in incremental rhetorical analysis. In *Proceedings of the 4th International Workshop on Linguistically Interpreted Corpora (LINC-03) at the EACL*, Budapest.

Stede, M. (2007). *Korpusgestützt Textanalyse. Grundzüge der Ebenen-orientierten Textlinguistik*. Gunter Narr, Tübingen.

Walsh, N. and Muellner, L. (1999). *DocBook: The Definitive Guide*. O'Reilly.

Webber, B. (1988). Discourse deixis: Reference to discourse segments. In *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistic (ACL'88)*, pages 113–122, Buffalo. State University of New York.

Webber, B. L. (1986). So what can we talk about now? In Grosz, B. J., Sparck Jones, K., and Webber, B. L., editors, *Readings in natural language processing*, pages 395–414. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Witt, A., Lüngen, H., Goecke, D., and Sasaki, F. (2005). Unification of XML documents with concurrent markup. *Literary and Linguistic Computing*, 20(1):103–116.

Wolf, F. and Gibson, E. (2006). *Coherence in Natural Language. Data Structures and Applications*. MIT Press, Cambridge, MA.

Zifonun, G., Hoffmann, L., and Strecker, B. (1997). *Grammatik der deutschen Sprache*, volume 7 of *Schriften des Instituts für deutsche Sprache*, chapter C6 "Thematische Organisation von Text und Diskurs", pages 535–591. de Gruyter, Berlin/New York.

Nils Diewald, Maik Stührenberg, Anna Garbar, Daniela Goecke

Serengeti – Webbasierte Annotation semantischer Relationen

Der Artikel stellt zum einen ein Annotationsschema für semantische Relationen vor, das für die Beschreibung eines deutschsprachigen Korpus für Training und Evaluation eines Systems zur Anaphernauflösung entwickelt wurde, zum anderen wird das webbasierte Annotationstool Serengeti beschrieben, das zur Annotation anaphorischer Relationen im Projekt A2 „Sekimo“ eingesetzt wird.[1] Im Gegensatz zu anderen Annotationstools benötigt Serengeti keine lokale Installation, was den Einsatz an einer großen Anzahl von Rechnern erleichtert. Darüber hinaus implementiert Serengeti ein Mehrbenutzerkonzept, das sowohl Gruppen als auch einzelne Nutzer unterstützt und zugehörige Dateien und Annotationen verwaltet.

1 Einleitung

In der Computerlinguistik und Sprachverarbeitung werden in verschiedenen Bereichen große Korpora qualitativ hochwertig annotierter Texte benötigt. Deren wachsende Bedeutung für die empirische Forschung, Hypothesentests sowie Training und Evaluation von Algorithmen maschinellen Lernens wird allgemein anerkannt. Um sowohl an Qualität wie an Quantität bestmögliche Ergebnisse zu erzielen, sind neben Annotationsschemata mit strikter Taxonomie und möglichst eindeutiger Interpretation einfach handhabbare Werkzeuge zur Annotation und Organisation der Korpora nötig, da sich die Erstellung einer empirischen Basis gerade im Forschungsgebiet der Anaphernresolution aufgrund der manuellen Annotation als hochgradig aufwändig erwiesen hat. Darüber hinaus kann auf Grund von Formatinkompatibilitäten selten auf bereits vorhandene Korpora zurückgegriffen werden. Insbesondere für die Erstellung von Korpora längerer Texte nimmt somit der Aspekt der nachhaltigen Korpuserstellung eine entscheidende Rolle ein.

Der Artikel gliedert sich wie folgt: Zunächst wird das Projekt „Sekimo“ vorgestellt (Abschnitt 2), um im Folgenden auf die darin geleistete Korpusarbeit, insbesondere in Bezug auf das zu Grunde liegende Annotationsschema und das verwendete Annotationsformat, einzugehen (Abschnitt 3). Das Annotationswerkzeug Serengeti wird in Abschnitt 4 vorgestellt, Abschnitt 5 behandelt aktuelle Entwicklungen hinsichtlich des generischen Repräsentationsformats SGF. Der Artikel schließt mit einem Ausblick auf die Weiterentwicklung von Serengeti (Abschnitt 6).

[1] Die in diesem Artikel präsentierten Arbeiten wurden im Rahmen des Projekts A2 „Sekimo“ der von der Deutschen Forschungsgemeinschaft geförderten Forschergruppe 437 *Texttechnologische Informationsmodellierung* durchgeführt. Für das genannte Korpus wurden zum einen vom Projekt A2 gesammelte Texte und zum anderen vom Projekt C1 zur Verfügung gestellte Texte verwendet.

2 Das Projekt „Sekimo“

Das Projekt A2 „Sekimo“ befasst sich mit der Integration heterogener linguistischer Ressourcen zur texttechnologischen Modellierung, wobei der Begriff Heterogenität sich hierbei beispielsweise auf das Repräsentationsformat oder die Funktion bezieht. Anwendungsdomäne ist die automatische Analyse anaphorischer Relationen. Um heterogene Ressourcen nutzbar zu machen, kommt ein abstraktes Datenformat zum Einsatz (vgl. Simons et al., 2004), wobei im Rahmen des Projekts sowohl ein auf einer Prolog-Faktenbasis aufbauendes (vgl. Witt et al., 2005) als auch ein rein XML-basiertes Repräsentationsformat (vgl. Stührenberg et al., 2006) entwickelt wurde. Mechanismen zur Integration sind notwendig, da es schwierig ist, die Ausgaben verschiedener linguistischer Ressourcen miteinander zu kombinieren: abgesehen von der Problematik, dass die aus einem Verarbeitungsschritt resultierende Ausgabedatei in den seltensten Fällen als Eingabe für einen nachfolgenden Verarbeitungsschritt verwendet werden kann, ist die Unifikation verschiedener Ausgaben (d. h. die Zusammenführung in eine einzelne XML-Datei) – die erst eine Analyse von Beziehungen zwischen Ebenen ermöglicht – auf Grund von XML-Inkompatibilitäten (überlappenden Elementstrukturen) oftmals nicht möglich.

3 Annotation anaphorischer Relationen

Die Darstellung der Annotation anaphorischer Relationen im Projekt „Sekimo“ ist unterteilt in die Diskussion des Annotationsschemas und der formalen Repräsentation in Form des Annotationsformats. Das Korpus enthält 49 deutschsprachige Texte, die sowohl aus Fachliteratur als auch aus Tages- und Wochenzeitungen stammen, davon wurden 14 Texte vollständig in Bezug auf anaphorische Relationen annotiert. Für diese Texte wurden insgesamt 4323 anaphorische Relationen auf der Basis von 11740 Diskursentitäten (65203 Token) annotiert.

3.1 Das Annotationsschema

Zur Annotation anaphorischer Relationen existiert eine Reihe an Formaten, angefangen von UCREL (vgl. Fligelstone, 1992; Garside et al., 1997) über das SGML-basierte MUC Annotationsschema (vgl. Hirschmann, 1997) hin zu dem auf XML basierenden MATE/GNOME Schema (vgl. Poesio, 2004), um nur einige zu nennen. Das im Projekt „Sekimo“ verwendete Schema basiert auf einer Annotationsrichtlinie für Koreferenzstrukturen, die im Projekt B1 „HyTex“ erarbeitet wurde (vgl. Holler et al., 2004), und die eine Erweiterung bzw. Präzisierung des genannten MATE/GNOME Schemas für die Anwendungsdomäne Hypertextualisierung darstellt. Die Grundidee besteht darin, die Unterscheidung zwischen Kospezifikation (vgl. Sidner, 1979) und Koreferenz in der Annotation abzubilden (vgl. Holler-Feldhaus, 2004). Während zwei Ausdrücke nur dann koreferieren, wenn sie auf dieselbe Entität in der Welt verweisen, genügt für Kospezifikation, dass ein Ausdruck einen vorangegangenen Ausdruck sprachlich wieder aufgreift.

Für das Projekt A2 wurde dieses Schema erweitert, um die Annotation indirekter Anaphorik (Bridging-Relationen, vgl. Clark, 1977) zu erlauben, hier ist das Antezedens einer Anapher nicht explizit realisiert, sondern muss aus dem Kontext erschlossen werden. Sowohl bei Kospezifikation als auch bei indirekter Anaphorik besteht neben der textuellen Ebene die semantische Interpretation: sprachliche Ausdrücke führen neue Diskursentitäten (Diskursreferenten in der Terminologie von Karttunen, 1976) ein und können auf bereits eingeführte Diskursreferenten verweisen; zwischen Diskursreferenten können semantische Relationen bestehen. Ein weiteres Schema, das die Annotation mehrsprachiger Korpora (*cross-linguistic anaphoric annotation*) fokussiert, stellen Krasavina and Chiarcos (2007) vor. Im Gegensatz zu dem vorgestellten Schema wird hier jedoch keine explizite Unterscheidung von Koreferenz und Kospezifikation angenommen.

Das Annotationsschema liegt in Form eines Manuals für die Annotatoren (Goecke et al., 2007a) sowie als XML DTD und XML Schema (XSD) vor, wobei die technische Realisation die Basis für das Annotationswerkzeug Serengeti darstellt.

Ausgangspunkt für die Annotation anaphorischer Relationen ist eine mehrstufige Vorverarbeitung, die verschiedene heterogene linguistische Ressourcen integriert. Zunächst werden Texte mit einer logischen Dokumentstruktur versehen, die u. a. Absätze, Sprachinseln, Abbildungen und Listen markiert, zusätzlich werden durch den funktional-dependenziellen Parser-Tagger Machinese Syntax der Firma Connexor Oy morphologisch-syntaktische Informationen auf Wortebene annotiert. Um Primärdatenidentität für die Unifikation mit der nicht-annotierten Referenz zu gewährleisten, werden die Originalannotationen des Parser-Taggers für die nachfolgenden Annotationsschritte modifiziert.[2] Der Begriff „Primärdatenidentität“ bezeichnet die Identität der zu Grunde liegenden Texte auf der Ebene der Zeichen. Im folgenden Schritt werden diejenigen Elemente, die Teil einer semantischen Relation sein können, so genannte *Markables*, identifiziert (vgl. Müller and Strube, 2001).[3] Im Projekt „Sekimo“, das die Detektion anaphorischer Relationen behandelt, dienen alle sprachlichen Ausdrücke, die einen Diskursreferenten im Sinne von Kamp and Reyle (1993) in die Diskurs- bzw. Textrepräsentation einführen, als relevante Diskursentitäten und somit als Markables. Die Identifikation erfolgt automatisch auf Basis der vom Machinese Syntax annotierten Wortformen. Zunächst werden einfache Diskursentitäten markiert, d. h., Diskursentitäten, die durch eine einfache NP realisiert sind. Aufbauend auf diesen können auch komplexe Diskursentitäten (also NPs mit Präpositionalphrase oder NPs mit NP als Prämodifizierer) annotiert werden. NPs mit Relativsatz werden nicht als komplexe Diskursentitäten markiert. Jede Diskursentität ist mit einem dokumentweit eindeutigen Identifikator versehen. Die zu untersuchenden semantischen Relationen, die zwischen Diskursentitäten bestehen, klassifizieren wir als Relationen der Kospezifikation (direkter Anaphorik) und indirekter Anaphorik (Bridging-Relationen, vgl. Clark, 1977; Vieira

[2] Eine Unifikation ist in diesem Fall problemlos möglich, da es zwischen der logischen Dokumentstruktur und der Annotation durch den Machinese Syntax nicht zu Überlappungen kommen kann.

[3] Die hier vorgestellte Version von Serengeti verwendet Texte mit vorannotierten Markables; eine derzeit in der Erprobung befindliche Fassung unterstützt bereits den Einsatz unannotierter Texte und das Hinzufügen von Markables während der Annotation.

and Poesio, 2001). Beide Typen können jeweils in weitere sekundäre Relationstypen unterteilt werden. Die direkte Anaphorik wird im Annotationsschema unterteilt in die Untertypen *ident*, *namedEntity*, *propName*, *synonym*, *hyperonym*, *hyponym*, *addInfo*, *paraphrase*. Der Wert *ident* wird vergeben, wenn sich ein Pronomen auf eine NP oder eine NP auf eine rekurrente NP bezieht. Kospezifikation zwischen einer NP, die nicht vom Typ *namedEntity* ist, und sich auf eine NP vom Typ *namedEntity* bezieht, wird mit dem entsprechenden sekundären Relationstyp ausgezeichnet. Der Wert *propName* wird vergeben, wenn die anaphorische Diskursentität ein Eigenname ist, und auf eine nominale Bezugsgröße im vorangegangenen Kontext verweist. Synonymiebeziehungen werden als solche markiert, wenn sich die Kopfnomen von Anapher und Antezedens in einer solchen befinden. Dabei ist zu beachten, dass im Projekt „Sekimo“ ein weiter Begriff der Synonymie verwendet wird, also der Kontext im Text entsprechend berücksichtigt wird, und auch Abkürzungen als Synonyme der jeweiligen Langform im Text ausgezeichnet werden. Hyperonymie und Holonymie zwischen den Kopfnomen von Anapher und Antezedens wird durch den entsprechenden sekundären Relationstyp ausgezeichnet. Bei den beiden Typen *addInfo* und *paraphrase* wird unterschieden, ob die kospezifierte NP neue oder zusätzliche Informationen einführt, bzw. die Anapher das Antezedens umschreibt.

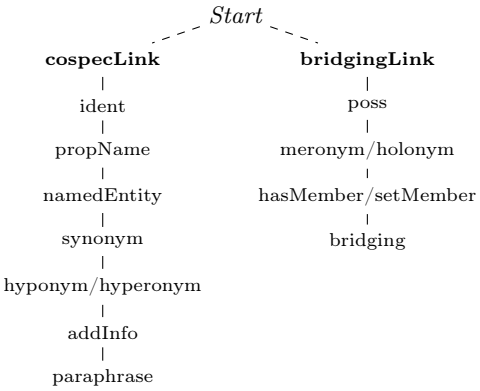


Abbildung 1: Der Entscheidungsbaum im Annotationsverlauf.

Analog zur Kospezifikation modelliert das vorliegende Annotationsschema auch Bridging-Relationen präziser. Dabei wurden die einzelnen Relationstypen so gewählt, dass sie durch linguistische Ressourcen (wie z. B. GermaNet, vgl. Goecke et al., 2007b) beschreibbar sind. Die Untertypen sind im Einzelnen: *poss*, *meronym*, *holonym*, *hasMember*, *setMember*, *bridging*. Als *poss* werden solche Relationen indirekter Anaphorik markiert, deren phorischer Ausdruck explizit durch ein Possessivpronomen oder eine Genitiv-NP besitzanzeigend markiert ist. Stehen Kopfnomen von Anapher und An-

tezedens in einer Meronymierelation, wird der entsprechende Wert genutzt, analog dazu die Holonymierelation. Davon abzugrenzen sind die Relationstypen *hasMember* und *setMember*. Ersterer liegt vor, wenn die Anapher eine Menge beschreibt, und das Antezendes ein Element dieser Menge; *setMember* wird als Relationstyp verwendet, wenn der phorische Ausdruck Element einer durch die Bezugsgröße beschriebenen Menge ist. Sollte keiner der genannten Relationstypen zutreffen, wird die allgemeine Relation *bridging* verwendet (z. B. Torte – Hochzeit). Eine weitere Unterteilung hinsichtlich Schema- oder Skriptbasierter Inferenz wird nicht vorgenommen. Zur Hilfestellung der Annotatoren bei der Entscheidung für einen Relationstyp wurde der in Abbildung 1 dargestellte Entscheidungsbaum entwickelt. Dabei können die Annotatoren den Entscheidungsbaum sequentiell überprüfen, d. h., nachdem sie die Entscheidung für Kospezifikation oder indirekte Anaphorik getroffen haben, können die einzelnen Subtypen nacheinander geprüft werden, wobei am Ende der Liste die allgemeinen Subtypen stehen, die nur gewählt werden sollten, sofern keiner der vorherigen Relationstypen als angemessen angesehen wurde. Darüber hinaus wurden Relationstypen definiert, die für Relationen gelten, deren Antezedentien durch nicht-nominale Einheiten eingeführt werden (z.B. Ereignisse, Fakten, Propositionen), und die wir der Terminologie von Asher (1993) folgend als *abstract event anaphora* bezeichnen. Für die Annotation von *abstract event anaphora* wurden drei Subtypen definiert: der Relationstyp *abstrProp* beschreibt anaphorische Relationen deren Antezdens durch eine Proposition eingeführt wird, Antezedentien des Relationstyps *abstrEvType* werden durch Ereignisse eingeführt und *abstrCluster* beschreibt diejenigen Relationen, deren Antezedens durch eine Summe von Propositionen bzw. durch einen Textabschnitt eingeführt wird.

3.2 Das Annotationsformat

Wie das MATE/GNOME-Schema ist das hier vorgestellte Annotationsformat XML-basiert und verwendet Standoff-Annotationen (vgl. Thompson and McKelvie, 1997), d. h. prinzipiell kann die Annotation unter Verwendung eines beliebigen XML-Editors durchgeführt werden. Listing 1 zeigt ein einfaches Beispiel aus dem Korpus, das mehrere Annotationsebenen enthält: die logische Dokumentstruktur (Element `para`), die Satzsegmentierung und Tokenisierung aus der Machinese Syntax-Ausgabe (Elemente `sentence` und `token`) sowie die Detektion der Diskursentitäten. Das Element `de`, das relevante Diskursentitäten markiert, trägt die drei obligatorischen Attribute `deID`, `deType` und `headRef`. Mittels `deID` kann über einen dokumentweit eindeutigen Wert jede Diskursentität identifiziert werden, `deType` gibt den Typ der Diskursentität (im Beispiel *namedEntity* oder *nom*) an und `headRef` referenziert das Kopfnomen der zu Grunde liegenden Nominalphrase (über XML ID/IDREF-Konstrukte). Token können Kindelemente der Elemente `de`, `sentence` oder `text` sein. Die diskurssemantischen Beziehungen werden als Kinder des Elements `standoff` gespeichert, hier finden sich auch weitere Informationen der Parser/Tagger-Ausgabe, die aus Gründen der Übersichtlichkeit ausgelagert wurden (Element `token_ref`).

Mittels des Elements `bridgingLink` wird indirekte Anaphorik zwischen zwei Diskursentitäten annotiert. Die in Abschnitt 3.1 genannten Subtypen werden dabei als Wert des Attributs `relType` spezifiziert, anaphorisches Element und Antezedens bzw. Antezedentien anhand ihrer dokumentweit eindeutigen ID in den Attributen `phorIDRef` und `antecedentIDRefs` referenziert. Analog dazu dient das Element `cospecLink` der Auszeichnung von Kospezfikation.

Listing 1: Das Annotationsformat für anaphorische Relationen.

```
<chs>
    <text>
        <para>
            <sentence>
                <de deID="de8" deType="namedEntity" headRef="w36">
                    <token ref="w36">Maik</token>
                </de>
                <token ref="w37">hat</token> <token ref="w38">kein</token>
                <token ref="w39">eigenes</token> <token ref="w40">Fahrrad</token>,
                <token ref="w42">und</token>
                <de deID="de10" deType="namedEntity" headRef="w43">
                    <token ref="w43">Marie</token>
                </de>
                <token ref="w45">fährt</token> <token ref="w46">nicht</token>
                <token ref="w47">in</token>
                <de deID="de11" deType="nom" headRef="w49">
                    <token ref="w48">den</token>
                    <token ref="w49">Urlaub</token>
                </de>.
            </sentence>
            <sentence>
                <de deID="de12" deType="nom" headRef="w53">
                    <token ref="w52">Zwei</token>
                    <token ref="w53">Kinder</token>
                </de>,
                <de deID="de13" deType="nom" headRef="w56">
                    <token ref="w55">eine</token>
                    <token ref="w56">Gemeinsamkeit</token>
                </de>:
            </sentence>
        </para>
    </text>
    <standoff>
        <token_ref id="w36" head="w37" pos="N" syn="@NH" depV="subj" morph="MSC␣SG␣
            NOM"/>
         [...]
        <semRel>
            <bridgingLink relType="hasMember" antecedentIDRefs="de8␣de10" phorIDRef="
                de12"/>
        </semRel>
    </standoff>
</chs>
```

Ambiguität wird durch die Definition mehrerer `cospecLink`- bzw. `bridgingLink`-Elemente realisiert, im Falle multipler Antezedentien wird auf mehrere Diskursentitäten in Antezedensposition verwiesen. Im Beispiellisting 1 besteht eine indirekt anaphorische Relation vom Typ *hasMember* zwischen den Antezedentien *Maik* (`de8`) und *Marie* (`de10`) und dem phorischen Ausdruck *Zwei Kinder* (`de12`).

Die Speicherung der semantischen Relationen als Standoff-Annotation am Ende der XML-Instanz ist der Tatsache geschuldet, dass die einzelnen Annotationsebenen jeweils durch eine entsprechende linguistische Ressource erstellt werden. Das allerdings erschwert die Verwendung eines einfachen XML-Editors zur Annotation, da oft zwischen verschiedenen Stellen im Dokument hin und her gewechselt werden muss. Aus diesem Grund wurde der Einsatz eines geeigneten Annotationswerkzeugs untersucht.

4 Serengeti

4.1 Annotationswerkzeuge

Für die Annotation unimodaler Daten sind in den letzten Jahrzehnten zahlreiche Werkzeuge entwickelt worden, die es dem Benutzer erleichtern, Texten Informationen hinzuzufügen. Neben für einen sehr begrenzten Einsatzbereich konzipierten Programmen, wie dem RST-Tool zur Erstellung von RST-Bäumen (vgl. O'Donnell, 1997), gibt es viele Annotationstools, die allgemeiner gestaltet sind und sich zur Beschreibung verschiedener semantischer Relationen in Texten eignen. Auch für die Annotation von Koreferenz existieren bereits spezialisierte Werkzeuge, wie XANADU (vgl. Garside and Rayson, 1997) oder der Coreferential Link Annotator (CLinkA, vgl. Orăsan, 2000). Die Vorteile, die eine Spezialisierung bietet, wie die optimierte Benutzerführung und die zielgerichtete Visualisierung, bringen jedoch auch Einschränkungen mit sich. So ist bei CLinkA das Annotationsschema direkt im Programm integriert und nicht erweiter- oder benutzerdefinierbar. Zugleich kann kein bereits auf einer anderen Beschreibungsebene annotierter Text verarbeitet werden. Dadurch lässt sich das Programm nur für sehr wenige Aufgaben einsetzen. Diese Beschränkungen führten zur Entwicklung des generalisierten Perspicuous and Adjustable Links Annotator (PALinkA, vgl. Orăsan, 2003), ein Werkzeug, das sich für unterschiedliche Schemata und Dokumente konfigurieren lässt. Solcherart generalisierte Annotationswerkzeuge haben den Vorteil, schnell an neue Aufgaben angepasst werden zu können. So muss der Benutzer nicht für jede neue Aufgabe den Umgang mit einem anderen Programm erlernen.

Zur Generalisierung verfolgen die Programme verschiedene Ansätze. MMAX (vgl. Müller and Strube, 2001), ein sehr populäres Tool zur Annotation von Koreferenz- und Bridgingbeziehungen, bietet eine große Funktionsvielfalt mit umfangreichen Möglichkeiten zur Definition des eigenen Annotationsschemas. WordFreak (vgl. Morton and LaCivita, 2003) hingegen bietet nur ein Basissystem, das durch Plugins, etwa die Integration automatischer Tagger, in seiner Funktionalität beliebig erweiterbar ist und es dem Nutzer erlaubt, sich seine persönliche Annotationsumgebung einzurichten. Weitere Annotationssysteme, wie GATE (vgl. Cunningham et al., 1996, 2002), bestehen

nicht aus einem einzigen Programm sondern aus einem Baukastensystem mit definierten Schnittstellen, durch die einzelne Programmmodule miteinander verbunden werden können.

Jeder dieser Generalisierungsansätze bedarf eines unterschiedlich großen Aufwands in Bezug auf die Konfiguration der Annotationsumgebung, um den eigenen Anforderungen zu genügen. Entweder müssen zu Beginn detaillierte Einstellungen bezüglich des Eingabe- und Ausgabeformats sowie des Annotationsschemas vorgenommen oder verschiedene Zusatzpakete zum Kernprogramm installiert werden. Da an der Erstellung eines Annotationskorpus unter Umständen viele Annotatoren beteiligt sind, die diese Vorgaben zu Beginn umsetzen müssen, kann der Konfigurationsaufwand erheblich sein. Ändert sich während der Arbeit das Annotationsschema – was insbesondere während der Entwicklungs- und Evaluationsphase nicht selten vorkommt – muss jeder Annotator diese Änderungen an seinem Programm vornehmen. Um dies zu vereinfachen, ist ein Konzept für kollaboratives Arbeiten vonnöten. Ein webbasiertes System beschränkt die Installation und Konfiguration der Umgebung auf einen Computer, den Web-Server. Beliebig viele Annotatoren können auf diese Umgebung zugreifen, ohne selbst aufwändige Konfigurationen vornehmen zu müssen; Änderungen des Annotationsschemas oder der Programmumgebung müssen nicht an mehreren Systemen vorgenommen werden. Auch für die Korpushaltung ist ein zentrales Konzept von Vorteil, um ortsungebunden und jederzeit auf das Korpus zugreifen zu können. Zudem ermöglicht es Annotatoren, zeitgleich an identischen Dokumenten zu arbeiten, unabhängig von ihrem Standort. Das Annotation Graph Toolkit (AGTK, vgl. Maeda et al., 2001; Ma et al., 2002) bietet die Möglichkeit, durch ein Client-Server-Modell mit mehreren Annotatoren an einer Annotation zu arbeiten. Das Annotat wird hierbei zentral in einer Server-Datenbank verwaltet, die Annotationsumgebungen selbst bleiben allerdings lokal installiert.

Da im Projekt „Sekimo“ neben der Korpuserstellung viel Wert auf die Evaluation des Annotationsschemas gelegt wurde, verfolgen wir bei der zentralen Korpusverwaltung den Ansatz, mehreren Personen zu ermöglichen, unabhängig voneinander ein Dokument zu annotieren und erst in einem weiteren Schritt durch Vergleich und Unifikation von Annotationen – anstelle von gemeinschaftlicher Annotation – eine verbindliche Fassung (*Gold Standard*) zu erstellen. Auf diese Weise kann eine Evaluation des Schemas durch einen Inter-Annotator-Vergleich stattfinden.

4.2 Architektur

Serengeti ist eine webbasierte Client-Server-Applikation für den Mozilla Firefox Browser[4] zur Annotation semantischer Relationen in Texten. Die hier vorgestellte Version des Programms ist noch weitgehend spezialisiert, mit einem im Vergleich zu den vorangehend vorgestellten Systemen geringen Funktionsumfang, und wird aktuell zu einem konfigurier- und erweiterbaren System ausgebaut.

[4] Serengeti unterstützt Firefox ab Version 1.5; Die Browsersoftware ist frei verfügbar unter `http://www.mozilla.com/firefox/`.

Auf Client-Seite, zur Darstellung des grafischen Benutzerinterfaces, werden bewährte Web-Technologien verwendet (XHTML, CSS, Javascript), auf Serverseite wird Perl eingesetzt.

Die Kommunikation zwischen Client und Server wird dabei aufgabenbedingt unterschiedlich gelöst: Lade- und Speicheroperationen mit geringem Datentransfer (etwa dem Empfang der Dokumentlisten oder dem Speichern erstellter Relationen) werden mittels einer AJAX-Engine durchgeführt (Asynchronous JavaScript and XML, Garrett, 2005), während für Operationen mit umfangreichem Datentransfer, wie dem Rendern der Dokumente, auf das klassische, synchrone Modell in Verbindung mit eingebetteten Frames zurückgegriffen wird (s. Abb. 2).[5]

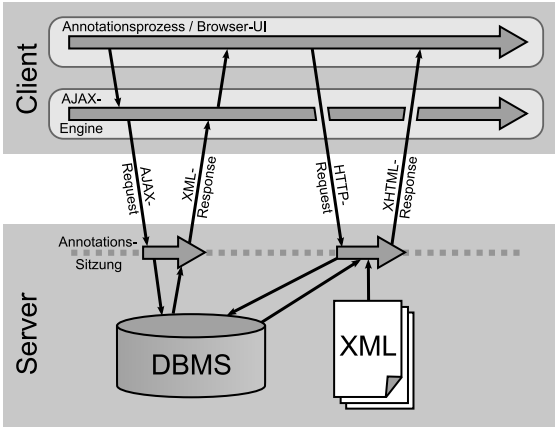


Abbildung 2: Client-Server-Kommunikation (vgl. Garrett, 2005)

Die zu annotierenden Dokumente sind als XML-Instanzen auf dem Server gespeichert und werden beim Aufruf in ein XHTML-Dokument transformiert. Die Annotationen, Projekt- und Benutzerdaten werden von einer MySQL-Datenbank verwaltet.
Die verteilte Architektur erlaubt es, Korpus- und Benutzerverwaltung serverseitig zu realisieren und die technischen Anforderungen auf Clientseite niedrig zu halten. Haben mehrere Personen ein und dasselbe Dokument annotiert, ermöglicht die zentrale Korpushaltung einen Vergleich und eine gesteuerte Unifikation beider Annotationen durch den Projektleiter.

4.3 Annotation mit SERENGETI

Nach der Anmeldung auf der Webseite[6] werden im oberen Teil der SERENGETI-Oberfläche zwei Menüs in Form von Auswahllisten eingeblendet (Gruppen- und Dokument-Menü, s. Abb. 3), durch die der Benutzer die Möglichkeit hat, das Annotationsprojekt sowie das zu annotierende Dokument auszuwählen.

Nach dem Laden des Dokuments kann unmittelbar mit der Annotation begonnen werden. Im oberen Abschnitt der grafischen Oberfläche, dem Text-Fenster, wird der zu annotierende Text visualisiert, der untere Abschnitt teilt sich in das Relations-Fenster auf der linken und das Editier-Formular auf der rechten Seite. Der Text wird mit Formatierungen bezüglich Paragraphen, Listen, Tabellen und nicht-textuellen Elementen

[5] Um Datenverlust zu vermeiden, sind während des Datentransfers allerdings keine weiteren Benutzeraktionen erlaubt, was dem „klassischen“ asynchronen AJAX-Ansatz widerspricht.
[6] Eine Demo-Installation ist unter `http://coli.lili.uni-bielefeld.de/serengeti/` zu finden.

dargestellt. Zudem sind alle Markables im Text durch Unterstriche markiert und mit ihrer eindeutigen ID ausgezeichnet, repräsentiert durch anklickbare Boxen, die es dem Annotator ermöglichen, die an einer semantischen Relation beteiligten Markables per Mausklick auszuwählen (oder gegebenenfalls zu verwerfen).

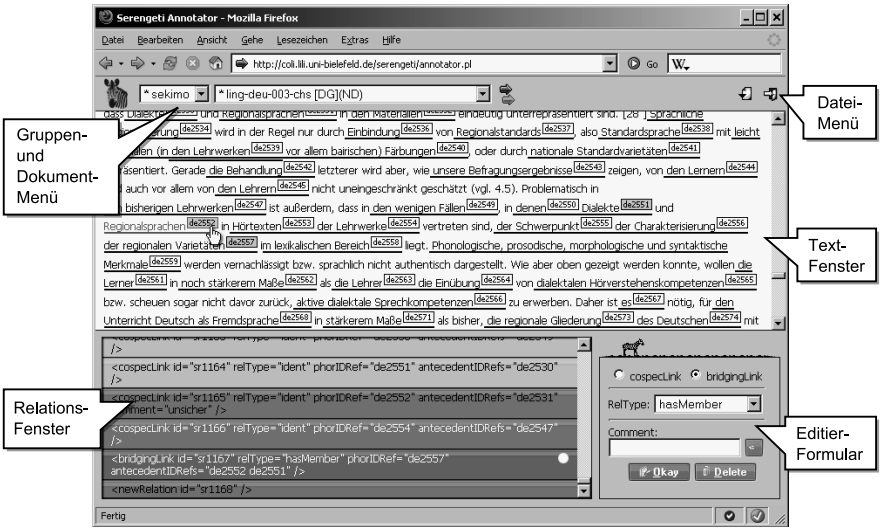


Abbildung 3: SERENGETI Hauptfenster

Für die Definition einer semantischen Relation werden im Rahmen des „Sekimo"-Projekts das anaphorische Element und ein oder mehrere Antezedentien markiert. Für die Annotation anaphorischer Relationen muss *zuerst* die Anapher und anschließend ein Antezedens bzw. mehrere Antezedentien ausgewählt werden. Um zwischen den beiden Typen der Diskursentitäten im Text zu unterscheiden, werden diese verschiedenfarbig (rosa – Anapher, blau – Antezedentien) dargestellt. Im nächsten Schritt wird die zwischen den beiden Diskursentitäten bestehende Relation definiert. Alle annotierten Beziehungen werden im Relations-Fenster als XML-Elemente gelistet. Am Anfang ist diese Liste bis auf einen gelben Balken leer, der die sogenannte `newRelation` enthält. Sie verbindet noch keine DEs und ist mit einem weißen Punkt versehen, der die aktuell ausgewählte Relation markiert.

Das Editier-Formular im rechten unteren Bereich des Fensters hält spezielle Optionen für die Erstellung und Bearbeitung von Relationen bereit. Im Falle der Annotation anaphorischer Relationen nach dem „Sekimo"-Annotationsschema (vgl. Abschnitt 3.1) soll zunächst der primäre Relationstyp bestimmt werden. Hier wird zwischen Kospezifikation und indirekter Anaphorik unterschieden. Dabei wird im Relations-Fenster nach Selektion des primären Typs der Elementname der `newRelation` zu `bridgingLink`

bzw. `cospecLink` geändert. Anaphern und Antezedentien werden durch die Attribute `phorIDRef` und `antecedentIDRefs` kodiert. Je nach ausgewähltem primären Typ ändert sich das Set der sekundären Relationstypen (vgl. Abb. 1), das im Editier-Formular als Auswahlliste dargestellt und dessen Wert vom Attribut `relType` übernommen wird.

Zusätzlich können Kommentare im Attribut `comment` gespeichert werden. Dies ist hilfreich, wenn der Annotator einen Vermerk zur annotierten Relation machen möchte. Solche Relationen werden grün eingefärbt. Nachdem eine Relation annotiert wurde, wird dies mit der Schaltfläche „Okay“ bestätigt. Bei vollständigen Relationen ohne Kommentare ändert sich die Farbe von Gelb auf Blau und eine neue `newRelation` wird im Relations-Fenster angelegt, welche als Nächstes bearbeitet werden kann. Ist nicht entscheidbar, auf welches Antezedens sich eine Anapher bezieht, können mehrere Relationen mit dieser Anapher definiert werden. Fehlerhafte Relationen können mit Hilfe des „Delete“-Buttons gelöscht werden. Diese Relationen werden zunächst rot hinterlegt und erst nach dem Speichern der Annotation endgültig aus der Liste entfernt. Unvollständig annotierte Relationen (z.B. in Bezug auf die teilnehmenden Diskursentitäten) werden in der Liste orangefarbig hervorgehoben und können später korrigiert werden.

Mit Hilfe des Datei-Menüs (s. Abb. 3) können Annotationen verwaltet werden, etwa durch Speichern, Drucken oder Exportieren. Die „View“-Option ermöglicht dem Annotator, abgeschlossene (d. h. weder kommentierte noch unvollständige) Relationen im Relations-Fenster auszublenden, um eine bessere Übersicht über die Annotation zu erhalten. Zum anderen können diejenigen Diskursentitäten, die bereits anaphorische Verwendung gefunden haben, durch die farbliche Hervorhebung ihrer ID-Boxen im Text-Fenster angezeigt werden.

4.4 Annotationsvergleich mit SERENGETI

Im so genannten *Consensus*-Modus besteht für bestimmte Mitglieder der Annotationsgruppe (die *Consensus-User*) die Möglichkeit, die Qualität der Annotationen mittels Inter-Annotator-Agreement zu verifizieren. Gleichermaßen lässt sich so auch das Schema überprüfen. Dieses Vorgehen hilft, die besten Annotationsergebnisse zu erzielen. Dabei werden zwei Annotationen im Relations-Fenster gleichzeitig dargestellt, wobei Relationen nach bestimmten Kriterien, etwa ihren anaphorischen Elementen, sortiert werden. In beiden Annotationen identische Relationen werden grau hinterlegt; in ausschließlich einer Annotation vorkommende erscheinen nur auf der entsprechenden Seite.

Falls Relationen lediglich ein Element (eine DE oder den Relationstyp) gemeinsam haben, werden sie einander gegenüber gestellt, wobei die Unterschiede hervorgehoben werden (s. Abb. 4: das anaphorische Element ist in beiden An-

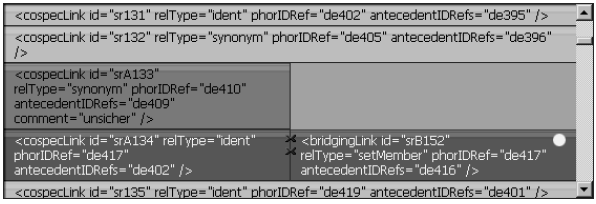


Abbildung 4: *Consensus*-Modus

notationen gleich, der Relationstyp und das Antezedens unterscheiden sich). Die im *Consensus*-Modus dargestellten Relationen können wie Relationen im Annotations--Modus bearbeitet (d. h. entfernt, geändert oder bestätigt) und die Annotation ebenso gespeichert werden. Ist eine Vergleichs-Annotation widerspruchsfrei, kann diese an weiteren Vergleichen teilnehmen.

5 SGF – Sekimo Generic Format

Neben der Verwendung innerhalb des „Sekimo"-Projekts wird SERENGETI auch für andere Anwendungen eingesetzt: abgesehen von der Möglichkeit, die aktuelle Version zur Annotation von lexikalischen Ketten einzusetzen (für eine prototypische Implementation vgl. Stührenberg et al., 2007), werden Teile der Architektur im Rahmen einer Kooperation mit der Universität Essex für die Projekte „AnaWiki" (vgl. Poesio and Kruschwitz, 2008) und die „AnaphoricBank"[7] genutzt und erweitert (s. Abschnitt 6). Unter anderem zu diesem Zweck wurde das *Sekimo Generic Format* (SGF) als Austauschformat für eine generalisierte Version von SERENGETI entwickelt. Ein weiteres Einsatzgebiet ist die weitergehende Analyse von Zusammenhängen zwischen einzelnen Elementen verschiedener Annotationsebenen. Dazu können unterschiedliche Architekturen eingesetzt werden. Der bisher im Projekt „Sekimo" verfolgte Weg (vgl. Abschnitt 3.2) war der Einsatz einer Standoff-Annotation sowie die Verwendung einer Prolog-Faktenbasis (vgl. Witt et al., 2005). Dabei erlaubt die Prolog-Faktenbasis die Analyse von Beziehungen zwischen Elementen verschiedener Annotationsebenen und ermöglicht so Aufschlüsse über mögliche Zusammenhänge zwischen linguistischen Merkmalsstrukturen (vgl. Lüngen et al., 2008). Hierzu müssen die XML-Instanzen in das Prolog-Format überführt werden. Unifikationen, die in der Prolog-Faktenbasis durchgeführt werden, nutzen zum XML-Export *Milestones* bzw. *Fragments* (vgl. Sperberg-McQueen and Burnard, 2002; Witt, 2002; DeRose, 2004), um überlappende Elemente auszuschließen.

Das im folgenden vorgestellte alternative *Sekimo Generic Format* hingegen ist vollständig XML-basiert und somit unabhängig von Zwischenformaten und erlaubt für den gesamten Prozess der Verarbeitung die Nutzung von XML-Software. Grundlage dazu ist das Konzept des *Annotation Graph* (vgl. Bird and Liberman, 1999, 2001), der einen Zeit- bzw. Zeichenstrahl als Basis für die Alignierung von Annotationen an die zu annotierenden Daten nutzt[8] – im Gegensatz zum *OHCO*-Modell (vgl. Renear et al., 1996), das eine geordnete Hierarchie aus verschachtelten Elementen modelliert, die sich als Baum darstellen lässt. Der Grund für den Einsatz eines graphenbasierten Modells liegt in der Problematik, mittels OHCO-basierter Inline-Annotation multiple Annotationen im Sinne einer Markup-Unifikation miteinander in Beziehung zu setzen, da es hier zu Überlappungen zwischen Elementen aus verschiedenen Ebenen kommen kann, die in XML nicht gestattet sind. Entsprechende Arbeiten dazu finden sich neben SGF in den Standardisierungsbestrebungen des ISO/TC 37/SC4 mit dem *Linguistic Annotation*

[7] `http://www.anaphoricbank.org`

[8] Das Konzept des Annotation Graph nutzt gelabelte azyklische Digraphen zur Darstellung linguistischer Annotationen.

Framework (LAF) und dem *Graph-based Format for Linguistic Annotations* (GraF; vgl. Ide, 2006; Ide and Suderman, 2007; Ide and Romary, 2007) sowie auf nationaler Ebene unter anderem im Kooperationsprojekt C2 der SFBs 441, 538 und 632, „Nachhaltigkeit linguistischer Daten“ (vgl. u. a. Dipper et al., 2006; Wörner et al., 2006; Eckart, 2006; Teich and Eckart, 2007; Witt et al., 2007).

Das Konzept der Datenhaltung von SGF sieht vor, alle zu einem Primärdatum zugehörigen Annotationen in einer Instanz zu speichern (im Gegensatz zu den anderen genannten Architekturen). Eine SGF-Instanz kann sowohl im Dateisystem (als Datei), in einer nativen XML-Datenbank, als auch in einer relationalen Datenbank oder einem hybriden Datenbanksystem gespeichert werden.[9]

Prinzipiell ist das Format sowohl zur Speicherung von textuellen als auch multimodalen Primärdaten nebst Annotationen geeignet und kann damit zur Analyse beliebiger linguistischer Phänomene herangezogen werden.[10] SGF ist vollständig XML-Schema--basiert und nutzt XML Namespaces (vgl. Bray et al., 2006) zur Trennung der einzelnen Annotationsebenen. Eine SGF-Instanz besteht immer aus dem *Base Layer* mit dem Namespace `http://www.text-technology.de/sekimo` und dem Präfix `base`, der grundlegende Funktionalitäten, Elemente und Attribute zur Verfügung stellt. Darüber hinaus kann eine beliebige Anzahl an Annotationsebenen, die jeweils eigenen XML-Namespaces zugeordnet werden, durch die `import`-Funktionalität in das Basis-Schema integriert werden (vgl. Thompson et al., 2004). Zur Validierung der jeweiligen Annotationsebenen können die ursprünglichen Dokumentgrammatiken (sofern sie als XSD vorliegen) genutzt werden, da das Basis-Schema sowohl für Metadaten als auch für Kindelemente des `layer`-Elements Konstrukte aus anderen Namensräumen zulässt. Listing 2 zeigt die nach SGF konvertierte Beispielannotation aus Listing 1.

Das Wurzelelement `corpus`, das mit einer eindeutigen ID und dem Korpustyp (*text* oder *multimodal*) versehen ist, umfasst ein oder mehrere `corpusdata`-Elemente. Im Kindelement `primaryData` können textuelle Primärdaten direkt gespeichert werden (bei kürzeren Texten, als Inhalt des `textualContent`-Elements) oder es wird mittels des Attributs `uri` des `location`-Elements auf eine externe Datei referenziert. Die Attribute `start` und `end` speichern den Wert des ersten bzw. letzten Zeichens (sofern es sich um einen Text handelt, sonst die Start- und Endzeit) der Primärdaten. Dabei wird jedes Zeichen, also auch Whitespaces (Leerzeichen, Umbrüche, Tabstops etc.) gezählt, empfehlenswert ist daher eine vorherige Normalisierung der Primärdaten in Bezug auf solche Zeichen. Es besteht die Möglichkeit, mittels des optionalen Elements `checksum` eine Prüfsumme für die Primärdaten zu speichern (im Listing 2 nicht gezeigt), die gewährleistet, dass externe Ressourcen auf dem gleichen Eingabetext arbeiten. Optionale Metadaten (Element `meta`, im Beispiel nicht enthalten) können dem gesamten Korpus

[9] Aktuelle Entwicklungsstufen von Serengeti nutzen ein SGF-API (Application Programming Interface), dem die Abbildung von SGF auf ein relationales Datenbanksystem (z.B. MySQL) zu Grunde liegt.

[10] Bei multimodalen Primärdaten wird an Stelle des Zeichenstrahls ein Zeitstrahl zur Alignierung der Annotationen genutzt. Die Verwendung multipler Primärdaten (z. B. einer Video- und einer Audiospur) ist möglich, allerdings muss ein Primärdatum ausgewählt werden, das den Zeitstrahl vorgibt.

Listing 2: SGF-Instanz (Ausschnitt)

```
<base:corpus  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://www.text-technology.de/sekimo"
    xmlns:base="http://www.text-technology.de/sekimo">
    <base:corpusData  xml:id="c1" type="text" sgfVersion="1.0">
        <base:primaryData  start="0" end="100" xml:lang="de">
            <base:location  uri="c1.txt"/>
        <base:primaryData>
        <base:segments>
            <base:segment  xml:id="seg1" start="0" end="100"/>
            <base:segment  xml:id="seg2" start="0" end="67"/>
            <base:segment  xml:id="to1" type="char" start="0" end="4"/>
            <base:segment  xml:id="to13" type="char" start="68" end="72"/>
            <base:segment  xml:id="seg5" type="seg" segments="to13␣to14"/>
        </base:segments>
        <base:annotation>
            <base:level  xml:id="doc" priority="0">
                <base:layer  xmlns:doc="http://www.text-technology.de/sekimo/doc">
                    <doc:text  base:segment="seg1">
                        <doc:para  base:segment="seg1"/>
                    </doc:text>
                </base:layer>
            </base:level>
        </base:annotation>
        <base:annotation>
            <base:level  xml:id="cnx" priority="0">
                <base:layer  xmlns:cnx="http://www.text-technology.de/cnx">
                    <cnx:sentence  id="w35" base:segment="seg2">
                        <cnx:token  base:segment="to1" xml:id="w36" head="w37" pos="N"
                              syn="@NH"
                                                depV="subj" morph="MSC␣SG␣NOM"/>
                    </cnx:sentence>
                </base:layer>
            </base:level>
        </base:annotation>
        <base:annotation>
            <base:level  xml:id="de" priority="1">
                <base:layer  xmlns:chs="http://www.text-technology.de/sekimo/chs">
                    <chs:de  base:segment="to1" deID="de8" deType="namedEntity"
                         headRef="w37"/>
                    <chs:de  base:segment="to7" deID="de10" deType="namedEntity"
                         headRef="to1"/>
                    <chs:de  base:segment="seg5" deID="de12" deType="nom" headRef="w53
                         "/>
                </base:layer>
            <base:level  xml:id="chs" priority="1">
                <base:layer  xmlns:chs="http://www.text-technology.de/sekimo/chs">
                    <chs:semRel>
                        <chs:bridgingLink  xml:id="sr1" relType="hasMember" phorIDRef=
                             "de12"
                                                antecedentIDRefs="de8␣de10"/>
                    </chs:semRel>
                </base:layer>
            </base:level>
        </base:annotation>
    </base:corpusData>
</base:corpus>
```

(als Kindelement von `corpus`), einzelnen Korpuseinträgen (unterhalb von `corpusData`) oder einer Annotationsebene (als Kindelement von `level`) zugeordnet werden, im Projekt „Sekimo“ werden hierzu Metadaten der „Open Language Archives Community“ (vgl. Simons and Bird, 2003) verwendet.

Da Annotationen am Zeichenketten- bzw. Zeitstrahl aligniert werden, werden für jede Annotationsebene Segmentierungen vorgenommen (`segments`). Dabei sollten neue Segmente (`segment`) nur dann hinzugefügt werden, wenn ein Element mit den entsprechenden Start- und Endpositionen nicht bereits durch Annotationen einer anderen Ebene eingeführt wurde. Da jedes Segment durch das Attribut `xml:id` eindeutig identifizierbar ist, kann im Anschluss der Segmentierung entsprechend darauf verwiesen werden. Eine Besonderheit stellt das Segment 'seg5' in Zeile 13 in Listing 2 dar: es besteht aus zwei Segmenten, womit eine hierarchische Beziehung zwischen Segmenten kodiert werden kann, die auch überlappende Segmente erlaubt. Jedes `corpusData`-Element kann eine Reihe von `annotation`-Kindelementen beinhalten. Dabei steht jedes `annotation`-Element für eine Annotationseinheit, innerhalb derer eine oder mehrere Annotationsebenen stehen dürfen – wobei das Element `level` die konzeptuelle Ebene der Annotation und das Element `layer` die XML-Realisierung speichert. Die Unterscheidung wird deutlich beim Vergleich der Annotationen, die jeweils die Ebene *doc* (logische Dokumentstruktur) bzw. *cnx* (Parser/Tagger-Ausgabe) beinhalten, mit der Annotation, die sowohl die Ebene *de* (Ebene der Diskursentitäten) als auch *chs* (Ebene der semantischen Relationen) beinhaltet.

Innerhalb eines `layer`-Elements sind die modifizierten Annotationen aus dem Ursprungsdokument enthalten. Dabei werden die Elemente wie folgt geändert: Elemente mit textuellem Inhalt (`PCDATA`) werden in leere Elemente überführt, Elemente mit gemischtem Inhaltsmodell werden zu reinen Container-Elementen (d. h. ohne *mixed content*). Elemente, deren Inhaltsmodell bisher nur aus anderen Elementen bestand, bleiben unverändert. So bleibt insbesondere die Hierarchiebeziehung zwischen Elementen einer Annotationsebene weiterhin direkt kodiert. Die Attribute bleiben ebenfalls unverändert – allerdings wird jedem Element das Attribut `segment` aus dem *Base Layer* hinzugefügt. Die im Listing 1 noch vorhandene Auslagerung der Token-Informationen mittels `token_ref` ist unnötig.

Relationen zwischen Elementen verschiedener Annotationsebenen lassen sich durch XPath- bzw. XQuery-Ausdrücke (vgl. Berglund et al., 2007; Boag et al., 2007) identifizieren. Im Verbund mit einer nativen XML-Datenbank oder einem hybriden Datenbanksystem lassen sich entsprechend umfangreiche Abfragen realisieren – aber auch auf Dateiebene lassen sich solche mit geeigneten XQuery-Prozessoren wie z. B. Saxon[11] durchführen. Eine ausführlichere Darstellung des Formats inklusive Evaluation ist in Stührenberg and Goecke (2008) gegeben.

[11] `http://saxon.sourceforge.net` bzw. `http://www.saxonica.com`

6 Zusammenfassung und Ausblick

Die in diesem Artikel vorgestellte Version des webbasierten Annotationssystems SERENGETI bietet bereits eine Reihe hilfreicher Werkzeuge zur Annotation semantischer Relationen und grenzt sich aufgrund seiner Architektur von vergleichbaren Werkzeugen ab. Das zu Grunde liegende Annotationsschema hat sich als sinnvolle Basis für die bisherige Annotationsarbeit erwiesen.

Im Zuge der aktuellen Generalisierung, zu der die Nutzung einer auf dem *Sekimo Generic Format* beruhenden Datenbank ebenso gehört wie die Möglichkeit, Markables während der Annotation hinzuzufügen und zu editieren, werden sowohl auf Client- als auch auf Serverseite Schnittstellen für Plugins etabliert. Diese erlauben eine Erweiterung der Werkzeugpalette sowie die Anpassung der Arbeitsumgebung an die Erfordernisse weiterer Annotationsaufgaben. Hierbei wird es möglich sein, beliebige Typen von Relationen und Markables für neue Annotationsprojekte zu definieren und für beliebige SGF-Layer Transformationsfilter zu ergänzen, die die HTML-Ausgabe steuern. Des Weiteren sind zusätzliche Funktionen für den Inter-Annotator-Vergleich geplant, etwa die automatische Berechnung von Übereinstimmungswerten.

Literatur

Asher, N. (1993). *Reference to abstract objects in discourse*, volume 50 of *Studies in Linguistics and Philosophy*. Kluwer Academic Publishers, Dordrecht, London, Boston.

Berglund, A., Boag, S., Chamberlin, D., Fernández, M. F., Kay, M., Robie, J., and Siméon, J. (2007). XML Path Language (XPath). Version 2.0. W3C Recommendation, World Wide Web Consortium.

Bird, S. and Liberman, M. (1999). Annotation graphs as a framework for multidimensional linguistic data analysis. In *Proceedings of the Workshop "Towards Standards and Tools for Discourse Tagging"*, pages 1–10. Association for Computational Linguistics.

Bird, S. and Liberman, M. (2001). A Formal Framework for Linguistic Annotation. *Speech Communication*, 33(1–2):23–60.

Boag, S., Chamberlin, D., Fernández, M. F., Florescu, D., Robie, J., and Siméon, J. (2007). XQuery 1.0: An XML Query Language. W3C Recommendation, World Wide Web Consortium.

Bray, T., Hollander, D., Layman, A., and Tobin, R. (2006). Namespaces in XML 1.0 (2nd Edition). W3C Recommendation, World Wide Web Consortium.

Clark, H. (1977). Bridging. In Johnson-Laird, P.N. & Wason, P., editor, *Thinking: Readings in Cognitive Science*, pages 411–420. Cambridge University Press, Cambridge.

Cunningham, H., Maynard, D., Bontcheva, K., and Tablan, V. (2002). GATE: An Architecture for Development of Robust HLT Applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*, pages 168–175. ACL.

Cunningham, H., Wilks, Y., and Gaizauskas, R. J. (1996). GATE – a General Architecture for Text Engineering. In *Proceedings of the 16th Conference on Computational Linguistics*, Copenhagen. COLING.

DeRose, S. J. (2004). Markup Overlap: A Review and a Horse. In *Proceedings of Extreme Markup Languages*.

Dipper, S., Hinrichs, E., Schmidt, T., Wagner, A., and Witt, A. (2006). Sustainability of Linguistic Resources. In Hinrichs, E., Ide, N., Palmer, M., and Pustejovsky, J., editors, *Proceedings of the LREC 2006 Satellite Workshop on "Merging and Layering Linguistic Information"*, Genua.

Eckart, R. (2006). Towards a modular data model for multi-layer annotated corpora. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 183–190, Sydney, Australia. Association for Computational Linguistics.

Fligelstone, S. (1992). Developing a Scheme for Annotating Text to Show Anaphoric Relations. In Leitner, G., editor, *New Directions in English Language Corpora: Methodology, Results, Software Developments*, pages 153–170. Mouton de Gruyter, Berlin.

Garrett, J. J. (2005). *AJAX: A New Approach to Web Applications*. Adaptive Path LLC. Online: `http://www.adaptivepath.com/publications/essays/archives/000385.php`.

Garside, R., Fligelstone, S., and Botley, S. (1997). Discourse Annotation: Anaphoric Relations in Corpora. In Garside, R., Leech, G., and McEnery, A., editors, *Corpus Annotation: Linguistic Information from Computer Text Corpora*, pages 66–84. Addison-Wesley Longman, London.

Garside, R. and Rayson, P. (1997). Higher-level annotation tools. In Garside, R., Leech, G., and McEnery, A., editors, *Corpus Annotation: Linguistic Information from Computer Text Corpora*, pages 179–193. Addison-Wesley Longman, London.

Goecke, D., Stührenberg, M., and Holler, A. (2007a). Koreferenz, Kospezifikation und Bridging: Annotationsschema. Interne Reports der DFG-Forschergruppe 437 "Texttechnologische Informationsmodellierung".

Goecke, D., Stührenberg, M., and Wandmacher, T. (2007b). Extraction and representation of semantic relations for resolving definite descriptions. extended abstract. In Mönnich, U. and Kühnberger, K.-U., editors, *OTT'06. Ontologies in Text Technology: Approaches to Extract Semantic Knowledge from Structured Information*, volume 1-2007 of *Publications of the Institute of Cognitive Science (PICS)*, pages 27–32. Institute of Cognitive Science, Osnabrück.

Hirschmann, L. (1997). MUC-7 Coreference Task Definition (version 3.0). In Hirschman, L. and Chinchor, N., editors, *Proceedings of Message Understanding Conference (MUC-7)*.

Holler, A., Maas, J.-F., and Storrer, A. (2004). Exploiting Coreference Annotations for Text-to-Hypertext Conversion. In *Proceedings of the 4th International Conference on Language Resources and evaluation (LREC 2004)*, volume II, pages 651–654, Lisbon, Portugal.

Holler-Feldhaus, A. (2004). Koreferenz in Hypertexten: Anforderungen an die Annotation. *Osnabrücker Beiträge zur Sprachtheorie (OBST)*, pages 9–29.

Ide, N. (2006). ISO/TC 37/SC4 N311: Linguistic Annotation Framework. Technical report, ISO/TC 37/SC4.

Ide, N. and Romary, L. (2007). Towards International Standards for Language Resources. In Dybkjaer, L., Hemsen, H., and Minker, W., editors, *Evaluation of Text and Speech Systems*, pages 263–284. Springer.

Ide, N. and Suderman, K. (2007). GrAF: A Graph-based Format for Linguistic Annotations. In *Proceedings of the Linguistic Annotation Workshop*, pages 1–8, Prague, Czech Republic. Association for Computational Linguistics.

Kamp, H. and Reyle, U. (1993). *From Discourse to Logic*. Kluwer: Dordrecht.

Karttunen, L. (1976). Discourse Referents. *Syntax and Semantics: Notes from the Linguistic Underground*, 7:363–385.

Krasavina, O. and Chiarcos, C. (2007). PoCoS - Potsdam Coreference Scheme. In *Proceedings of the Linguistic Annotation Workshop*, pages 156–163, Prague, Czech Republic. Association for Computational Linguistics.

Lüngen, H., Bärenfänger, M., Goecke, D., Hilbert, M., and Stührenberg, M. (2008). Anaphoric relations as cues for rhetorical relations. erscheint in LDV Forum – Zeitschrift für Computerlinguistik und Sprachtechnologie.

Ma, X., Haejoong, L., Bird, S., and Maeda, K. (2002). Models and Tools for Collaborative Annotation. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, Paris. European Language Resources Association.

Maeda, K., Bird, S., Ma, X., and Lee, H. (2001). The Annotation Graph Toolkit: Software Components for Building Linguistic Annotation Tools. In *HLT '01: Proceedings of the first international conference on Human language technology research*, pages 1–6, Morristown, NJ, USA. Association for Computational Linguistics.

Morton, T. and LaCivita, J. (2003). WordFreak: An Open Tool for Linguistic Annotation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 17–18, Edmonton, Canada.

Müller, C. and Strube, M. (2001). Annotating Anaphoric and Bridging Relations with MMAX. In *Proceedings of the 2nd SIGdial Workshop on Discourse and Dialogue*, pages 90–95, Aalborg, Denmark.

O'Donnell, M. (1997). RST-Tool: An RST Analysis Tool. In *Proceedings of the 6th European Workshop on Natural Language Generation*, Duisburg, Germany.

Orăsan, C. (2000). CLinkA a Coreferential Links Annotator. In *Proceedings of the Second International Conference on Language Resources and Evaluation*, pages 491–496. LREC.

Orăsan, C. (2003). PALinkA: A highly customisable tool for discourse annotation. In *Proceedings of the Fourth SIGdial Workshop on Discourse and Dialogue*, Sapporo, Japan.

Poesio, M. (2004). The MATE/GNOME Scheme for Anaphoric Annotation, Revisited. In *Proceedings of SIGDIAL*, Boston.

Poesio, M. and Kruschwitz, U. (2008). ANAWIKI: Creating anaphorically annotated resources through web cooperation. Submitted to LREC 2008.

Renear, A., Mylonas, E., and Durand, D. (1996). Refining our notion of what text really is: The problem of overlapping hierarchies. *Research in Humanities Computing. Selected Papers from the ALLC/ACH Conference, Christ Church, Oxford, April 1992*, 4:263–280.

Sidner, C. (1979). *Towards a computational theory of definite anaphora comprehension in English discourse*. PhD thesis, MIT.

Simons, G. and Bird, S. (2003). *OLAC Metadata*. OLAC: Open Language Archives Community.

Simons, G., Lewis, W., Farrar, S., Langendoen, T., Fitzsimons, B., and Gonzalez, H. (2004). The Semantics of Markup. In *Proceedings of the ACL 2004 Workshop on RDF/RDFS and OWL in Language Technology (NLPXML-2004)*, Barcelona.

Sperberg-McQueen, C. and Burnard, L., editors (2002). *TEI P4: Guidelines for Electronic Text Encoding and Interchange*. published for the TEI Consortium by Humanities Computing Unit, University of Oxford, Oxford, Providence, Charlottesville, Bergen.

Stührenberg, M. and Goecke, D. (2008). SGF – an integrated model for multiple annotations and its application in a linguistic domain. In *Proceedings of Balisage: The Markup Conference*.

Stührenberg, M., Goecke, D., Diewald, N., Cramer, I., and Mehler, A. (2007). Web-based Annotation of Anaphoric Relations and Lexical Chains. In *Proceedings of the Linguistic Annotation Workshop (LAW)*, pages 140–147, Prag. Association for Computational Linguistics.

Stührenberg, M., Witt, A., Goecke, D., Metzing, D., and Schonefeld, O. (2006). Multidimensional Markup and Heterogeneous Linguistic Resources. In *Proceedings of the 5th Workshop on NLP and XML (NLPXML-2006): Multi-Dimensional Markup in Natural Language Processing*, pages 85–88.

Teich, E. and Eckart, R. (2007). An XML-based data model for flexible representation and query of linguistically interpreted corpora. In *Datenstrukturen für linguistische Ressourcen und ihre Anwendungen. Data Structures for Linguistic Resources and Applications. Proceedings of the Biennial GLDV Conference 2007*, Tübingen.

Thompson, H. S., Beech, D., Maloney, M., and Mendelsohn, N. (2004). XML Schema Part 1: Structures (2nd Edition). W3C Recommendation, World Wide Web Consortium.

Thompson, H. S. and McKelvie, D. (1997). Hyperlink semantics for standoff markup of read-only documents. In *Proceedings of SGML Europe '97: The next decade – Pushing the Envelope*, pages 227–229, Barcelona.

Vieira, R. and Poesio, M. (2001). An empirically based system for processing definite descriptions. *Computational Linguistics*, 26(4):539–593.

Witt, A. (2002). *Multiple Informationsstrukturierung mit Auszeichnungssprachen. XMLbasierte Methoden und deren Nutzen für die Sprachtechnologie*. Dissertation, Universität Bielefeld.

Witt, A., Goecke, D., Sasaki, F., and Lüngen, H. (2005). Unification of XML Documents with Concurrent Markup. *Literary and Lingustic Computing*, 20(1):103–116.

Witt, A., Schonefeld, O., Rehm, G., Khoo, J., and Evang, K. (2007). On the Lossless Transformation of Single-File, Multi-Layer Annotations into Multi-Rooted Trees. In *Proceedings of Extreme Markup Languages*, Montréal, Québec.

Wörner, K., Witt, A., Rehm, G., and Dipper, S. (2006). Modelling Linguistic Data Structures. In *Proceedings of Extreme Markup Languages*, Montréal, Québec.

Sonja Ruda

Model of a Teacher Assisting Feedback Tool for Marking Free Worded Exercise Solutions

Free worded exercise solutions involve the advantage that students entirely have to resort to their own knowledge. However, their disadvantage is that they cannot be corrected automatically – in contrast to exercises with preset solution possibilities, e.g. multiple choices. This paper outlines these two types of exercise solutions, the methods and some results of a pragmalinguistic analysis of exercise types and their solutions as well as teachers' correction actions of free worded exercise solutions. Afterwards a prototype of a feedback tool model, based on this study and assisting teachers' correction actions, will be briefly introduced.

1 Some Problems Involved in Exercise Solving and their Correction

Teaching and learning systems with implemented exercises can identify errors simply as formal deviations by comparing the learners' input with the solution preset by the teacher (cf. Klemm and Ruda 2003, 183–185; Klemm et al. 2004, 118–125; Narciss et al. 2004). So far texts of free worded solutions have either not been demanded or can merely be evaluated in the simplest way. The following four problems may result:

1. Terms which were not entered by the student in the designated order could make a wrong sense. If the teacher has not considered ambiguous word combinations, a false solution is though assessed as "correct" since the demanded terms are included.

2. If the teacher does not consider correct alternatives, right solutions can nevertheless be marked as "wrong".

3. If a student's answer is assessed as "not completely correct", the corresponding text passages cannot be marked exactly.

4. If the teacher offers sample solutions, he has no control whether the student compared the solutions correctly and if he understood why his own solution may be wrong. Although the teacher can monitor the students through their user profiles of a database, a large number of participants and exercises can make this supervision almost impossible.

Exercises with preset answers can also be corrected via computer without problems. However, there is a high probability that the students do not take them seriously enough, that they are quickly bored, even deliberately enter wrong answers or simply guess the

Sonja Ruda

Model of a Teacher Assisting Feedback Tool for Marking Free Worded Exercise Solutions

Free worded exercise solutions involve the advantage that students entirely have to resort to their own knowledge. However, their disadvantage is that they cannot be corrected automatically – in contrast to exercises with preset solution possibilities, e.g. multiple choices. This paper outlines these two types of exercise solutions, the methods and some results of a pragmalinguistic analysis of exercise types and their solutions as well as teachers' correction actions of free worded exercise solutions. Afterwards a prototype of a feedback tool model, based on this study and assisting teachers' correction actions, will be briefly introduced.

1 Some Problems Involved in Exercise Solving and their Correction

Teaching and learning systems with implemented exercises can identify errors simply as formal deviations by comparing the learners' input with the solution preset by the teacher (cf. Klemm and Ruda 2003, 183–185; Klemm et al. 2004, 118–125; Narciss et al. 2004). So far texts of free worded solutions have either not been demanded or can merely be evaluated in the simplest way. The following four problems may result:

1. Terms which were not entered by the student in the designated order could make a wrong sense. If the teacher has not considered ambiguous word combinations, a false solution is though assessed as "correct" since the demanded terms are included.

2. If the teacher does not consider correct alternatives, right solutions can nevertheless be marked as "wrong".

3. If a student's answer is assessed as "not completely correct", the corresponding text passages cannot be marked exactly.

4. If the teacher offers sample solutions, he has no control whether the student compared the solutions correctly and if he understood why his own solution may be wrong. Although the teacher can monitor the students through their user profiles of a database, a large number of participants and exercises can make this supervision almost impossible.

Exercises with preset answers can also be corrected via computer without problems. However, there is a high probability that the students do not take them seriously enough, that they are quickly bored, even deliberately enter wrong answers or simply guess the

answer if they do not know the correct solution – particularly if they just do self-tests (cf. Neumann, 2003, 22 p.) which the teacher never views.

Ingenkamp (1993, 200) objects this criticism on solution choice exercises by stating that the possible disadvantage of blind guessing is more than compensated for by the advantage of a considerably larger sample of possible questions. He points out that they are not generally easier than free answer forms but, depending on the offered answers, rather more difficult. According to him the offer of wrong or only partly right answers in connection with the request for correction or critical choice is doubtlessly justified and even desired from a didactical point of view. Moreover, answer choice exercises are compiled and subsequently tested according to a thoroughly elaborated system (Lienert and Raatz, 1998). This is to ensure that the learners need to have certain knowledge in order to solve the exercises correctly. Nevertheless self worded answers offer a considerably bigger learning effect since students entirely have to resort to their own knowledge. This is one of the reasons why multiple choice tasks are no longer permitted for certain examinations (Sächsisches Oberverwaltungsgericht Bautzen, 2002) thus not even necessarily being suitable any more for their preparation.

Furthermore e-learning students' answers are frequently assessed in a very simple way, e.g., "Your answer is unfortunately wrong". All this demonstrates technical shortcomings, the enormous implementation expenditure of a differentiated feedback for every single solution possibilities for all exercises of the (online) course and the involved costs. Therefore a teacher-assisting feedback tool for marking free worded exercise solutions could be helpful in solving these problems. This feedback tool shall connect the advantages of free worded exercise solutions with those processed by the computer.

2 Methods for Analyzing Free Worded Exercise Solutions and their Correction

The development of the feedback tool (Ruda, 2008) involved the classification of types of exercises and solutions. Suitable exercise types and solutions were determined including, e.g., questions which demand standardized answers and which should moreover be brief – i.e. interpretations and argumentations for example are not suitable – and which should follow the descriptive and explicative subject development pattern (cf. Brinker, 2005, 65–69 and 75–79).

Teachers' correction actions were determined by consulting a text corpus consisting of offline university examination questions which demand self worded answers with at least one word. I followed the communicative-pragmatic approach that, under certain circumstances, already considers a word or a sentence as a text (Brinker, 2005, 18). The research methods included the 'think aloud method' (van Someren et al., 1994) and the 'concurrent record analysis' in combination with the 'focused interview' (Karbach and Linster, 1990, 84 p. and 87). During the concurrent recording the expert solves a problem thinking *aloud*, i.e., he says what he sees, feels, thinks and does. Concurrent records document a correct but not complete trace through the problem solving process. All documented actions and observations were really carried out.

However usually some information on the expert's actions is missing. This can be obtained through certain questions in the focused interview.

14 lecturers produced 15 verbal records which were taped, transcribed and described with regard to actions thus providing analytical transcripts. The lecturers have each revised between three and seven papers. Altogether there are 71 papers from 13 special fields with 488 corrected exercises including subtasks. The tape recordings lasted from about nine minutes up to almost two hours. The average recording amounted to one hour.[1]

The correction actions included aspects like: How is the comparison between the lecturer's and student's solution text made? How are mistakes recognized, analyzed, corrected and assessed? Which actions are performed by all lecturers under consideration?

This analysis is based on Austin's (1962) and particularly Searle's (1969) 'Speech Act Theory' and the subsequent related works by Holly et al. (1984), von Polenz (1988) and Brinker (2005).

A deductive research method was employed since it is regarded to be general knowledge that the teacher revises and subsequently assesses students' solutions. The determination of particular (speech) actions that may be formalized is of special importance. The conversion of certain structures into the feedback tool, on the other hand, was based on the inductive approach since concrete and thus specialized examples were translated into general rules.

The intention to implement structures that can be formalized into the feedback tool yielded, amongst others, the following questions: Which actions and speech actions of the lecturers can be simulated by the feedback tool? Which assessments are unambiguous and which are vague? Which of the ambiguous forms can be imitated by the feedback tool? How is the interaction between teacher and feedback tool facilitated? Which suggestions are offered by the feedback tool to the teacher? Which questions have to be asked by the feedback tool? How can the feedback tool be designed in order not to overstrain but to support the teacher when generating correction and assessment standards? How can the feedback tool be designed that the teacher, despite accepting the preset correction and assessment patterns, has the last word in the final assessment? Which rules have to be followed by the teacher? What does the teacher have to know before starting to work with the feedback tool?

It was aimed to design a feedback *tool* which supports teachers in their correction work by making arrangements with them regarding the solution texts and assessments which requires the acquisition of certain rules and initial training. This means that an automatic correction of all solution texts cannot and should not be expected[2]. The feedback tool conception is not based on a flow model but on a scheme which reveals certain correction patterns originating from empirical results. This scheme is to be understood as an ideal type of a model according to Weber (1980, 4). It needs to be emphasized that it was not intended to implement a "phase model" which would force the teacher into a fixed corset

[1] For reasons of data protection students' answers as well as complete transcriptions are not disclosed.

[2] For the problems and complexity of text understanding and/or -production with computer cf. Winograd and Flores (1992), Rothkegel (1989) as well as Lobin and Lemnitzer (2004).

thus restricting him far too much in his action diversity and freedom (cf. Suchman 1987; Heath and Luff 2000, 9–12). It was rather aimed to implement a "task oriented model" (cf. Brünner, 2000, 27) providing eligible knowledge resources and thus supporting the teachers' current assessment actions in a flexible way.

3 Teachers' Actions of Correcting Free Worded Exercise Solutions

Correcting is often described with only few steps (Kleppin, 2007, 55). However, correcting involves many more actions than merely identifying, marking and possibly improving mistakes or grading. The analysis of the present transcripts has shown that correcting consists of five constitutive partial actions in the following sequence:

1. Grasping the complete student's solution attempt or filtering the propositions to be dealt with
2. Comparison with the expected solution
3. Assessment of the student's solution attempt with regard to the definition of the tasks
4. Total assessment of student's solution and
5. Total assessment of all present solutions of a student.

The potential correction actions also include preparations like writing down expected solutions, compiling a criteria grid and determining the approach – paper by paper or exercise by exercise. The envisioning of exercise questions and solutions can take place before the first, second or third partial action. The third step comprises corrections and comments. Remarks concerning the assessment in general can be made at the beginning and/or the end of the revision.

The most challenging action is the individual assessment of a student's exercise solution, which consists of seven areas:

1. Envisioning of the exercise definition
2. Envisioning of the connection between exercise definition and solution
3. Expressing the first impression
4. Grasping the student's solution attempts
5. Comparison with the expected solution
6. Assessing individual parts of the student's solution attempt and
7. Correcting.

It was revealed that there are complex considerations about correct and incorrect facts and forms mainly in the case of problematic and vague solution attempts, which the teachers tried to solve intuitively employing certain problem solving strategies (Ruda, 2008, 166–185).

A feedback tool can support the teacher in the complex action of correcting since this is effected according to a certain scheme. The constitutive individual assessment actions can be outlined as follows: The feedback tool captures the student's solution attempt, assigns it to the question, compares it with the stipulated solution possibilities and carries out assessments and, if necessary, corrections defined before by the teacher. Therefore the teacher is prompted to thoroughly study the different answer possibilities in advance, thus having more time afterwards to deal with the (problematic) vague solution attempts.

4 Model of a Teacher-Assisting Feedback Tool

Before the first application of the feedback tool, the teacher is given important use guidelines. This includes technical requirements, conceptual conditions for task setting, information on filtering relevant solution contents – according to the subject development patterns like describing (Brinker, 2005, 65–69) –, as well as the functionality and encoding of the offered operators.

4.1 Operators

The teacher arranges solution possibilities and assessments (cf. Figure 1).

First the teacher enters the identification and the text of the exercise. As the next step he sets assessment rules for text parts of the solution. There are two options: He can either encode the rules by himself, which requires exact knowledge and experience about the rules, or, if he lacks this knowledge, he can start the rule assistant. The rule assistant offers seven operators: Boole's operators AND, OR, NOT and, based on them, the operators AND POSSIBLE, OR POSSIBLE, POINT and PREFERRED. They result in pattern matching.

1. The most relevant operator is the AND (UND) Operator (cf. Figure 2). The teacher is asked to enter data, at least one word and/or phrase which have to be part of the solution. Different singular, plural and flexion forms will be considered by setting wildcards: *action*, *actions* → *action**.

2. The OR (ODER) operator (cf. Figure 3) is optional like the further operators. The number of words and phrases is at will again. The teacher can enter alternative solution possibilities. If a variant is preferred, it is marked by setting a tick with a mouse click thus activating the

3. PREFERRED (BEVorzugt) operator (cf. Figure 3): The other possible solution is marked as "not optimal" but "nevertheless correct". The feedback tool also displays the optimal solution in this case.

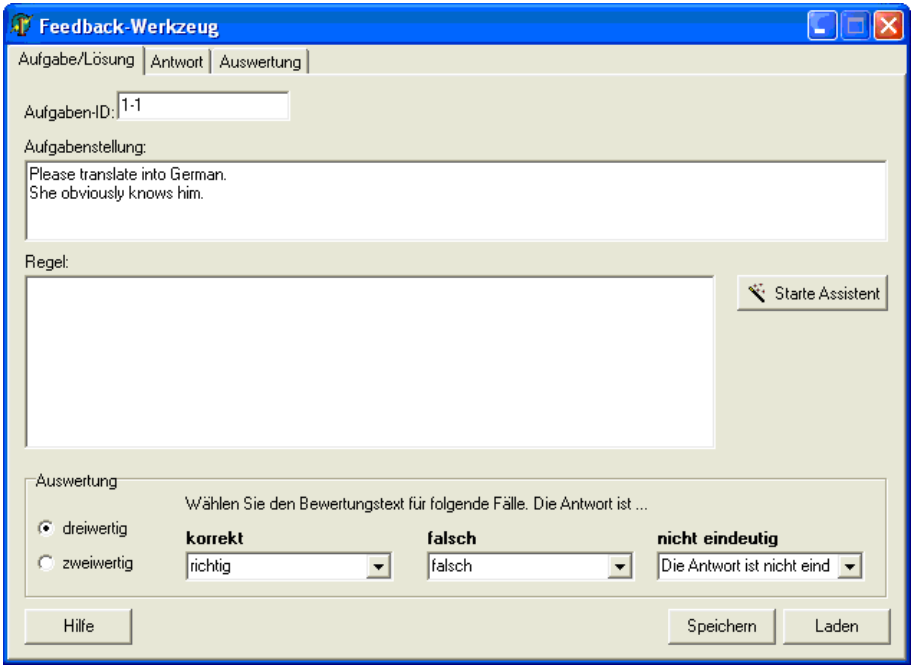


Figure 1: The feedback tool area “exercise/solution”.

4. OR POSSIBLE (ODEREVTL) offers the inclusion of solution possibilities which are not completely correct but nevertheless acceptable or which are assessed as “correct” although not necessarily corresponding with the expected expression.

5. With AND POSSIBLE (UNDEVTL) the teacher can display solution possibilities which are right or acceptable provided that they are given in addition to the AND-entry. However, as soon as they are isolated they are no longer assessed as absolutely correct since they may be too general.

6. In the mask POINT (PLUS) the teacher can enter solutions which are better than the expected solution and which are therefore assessed with “outstanding” or honoured with an extra point.

Figure 2: The AND mask.

7. The NOT (NICHT) operator includes expressions which must not be included in the solution. The teacher can enter expressions ranging from an assessment as simply wrong up to an evaluation as a grave mistake.

In the initial stage, the teacher should have all operators demonstrated in the fixed sequence. With some practice he can specifically control the corresponding operators through the file card symbols and finalize the process by clicking the button "apply" (übernehmen). The encoded rules can subsequently be viewed and, if necessary, changed in the field "Rules" (cf. Figure 4).

These operations enable the teacher not only to envision concrete solutions but also to realize incorrect and vague solutions. Thus the rule assistant facilitates the development of an assessment standard which is more extensive than usual. The stipulated rules guarantee a standardized and therefore objective assessment. At this stage the teacher is recommended to consider not only solution possibilities formulated by himself but to likewise consult some of the participants' solutions.

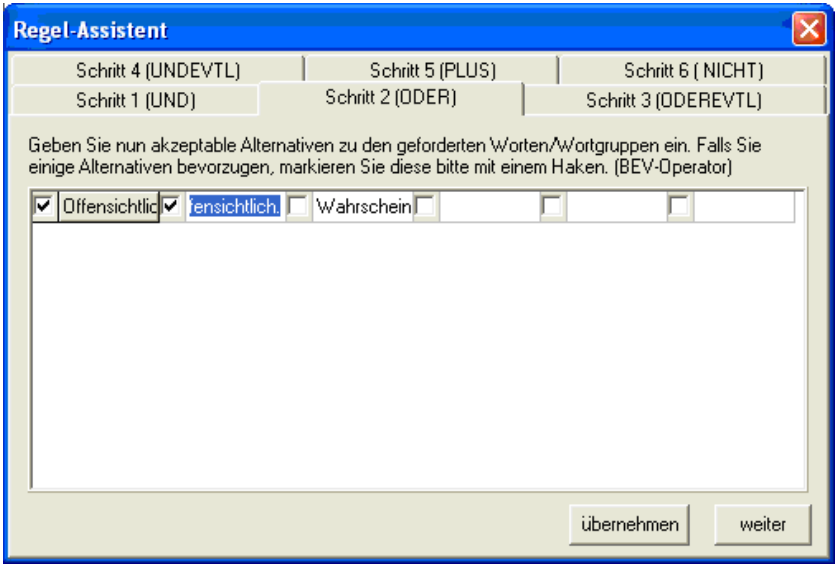


Figure 3: The OR and PREFER mask.

4.2 Evaluation Possibilities and Assessment Texts

Then the teacher has to decide between the two- and three-valued evaluation (cf. Figure 1). The two valued modus merely concerns the categories “right” and “wrong” and is useful if there are only few correct solutions while all the others can be definitely qualified as false. For all other cases the three-valued modus also includes vague solutions. This

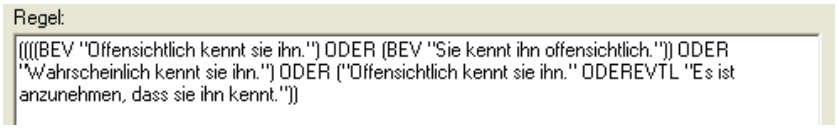


Figure 4: The rule field.

selection option is preset since it is mostly applicable with the present text corpus.

In the next step the teacher selects corresponding assessment texts (cf. Figure 1). He can either choose variants from the text corpus or enter his own wording.

4.3 Answer and Evaluation

The teacher copies the student's solution from a data base and pastes it into the answer mask. If, on the other hand, the feedback tool is integrated into a web based training platform, the student's answer appears directly in the answer mask. The feedback tool evaluates the student's answer in accordance with the agreed rules and shows it in the bottom field (cf. Figure 5).

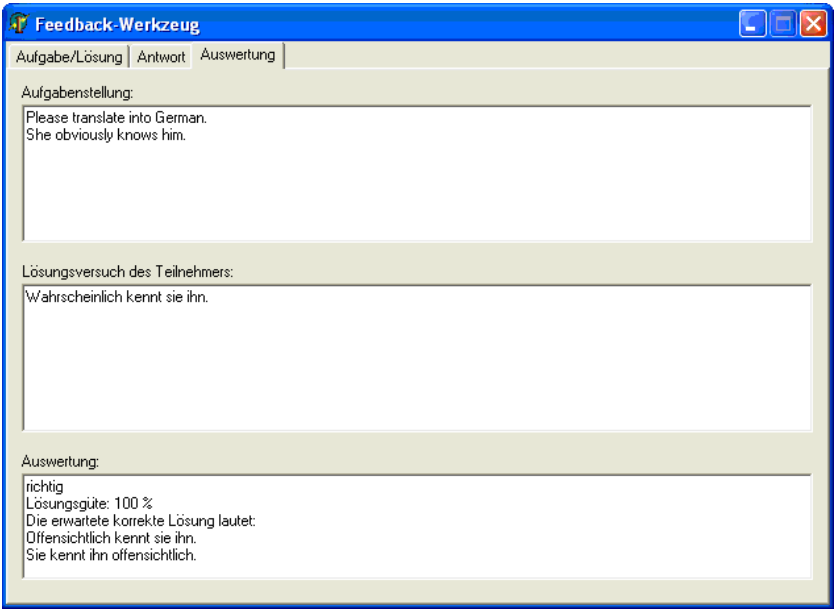


Figure 5: A correct, but not ideal solution.

The feedback tool displays the agreed assessment text, the solution quality in per cent and the expected correct solution since the students' solution concerning the adverb "wahrscheinlich" is not completely optimal. Since from the teacher's point of view this is not bad, the solution quality is nevertheless 100 % (cf. Figure 5).

Students' solution attempts which do not fit into the set framework are returned to the teacher by the feedback tool (cf. Figure 6). The teacher can correct them by hand and

then revise and complement the feedback tool with the corresponding rules and data. Therefore the output of non-revised solutions will be reduced with each learner group.

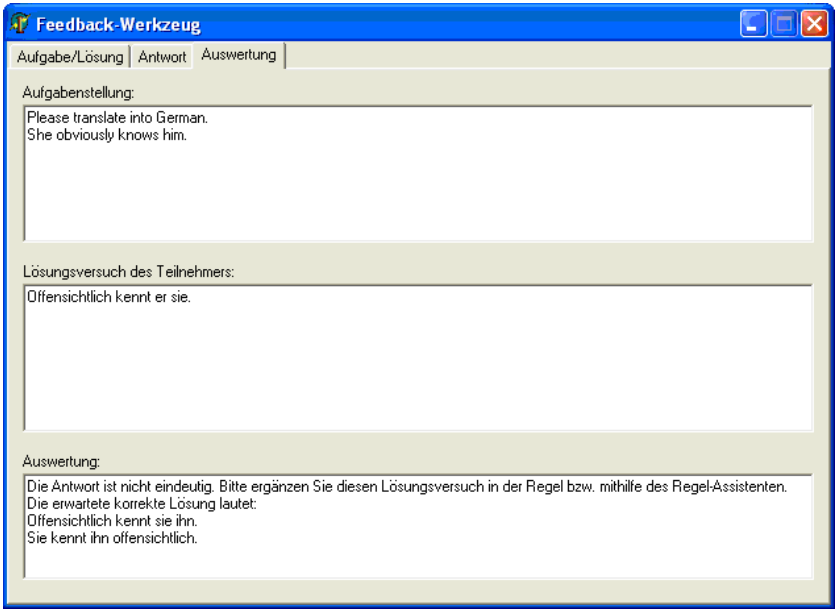


Figure 6: New answer possibility with interchanged pronoun.

5 Summary and Outlook

This feedback tool is oriented towards free worded exercise solutions whose texts are standardized and brief. It is designed for teachers with a large number of students thus having many exercise solutions attempts to correct.

With the feedback tool the teacher draws up possible answers ranging from *scrupulously correct* over *not quite right/false* up to *absolutely wrong*. Students' solution attempts which do not yet match the rules are returned by the feedback tool for further processing on the part of the teacher.

The feedback tool offers the teacher the following advantages:

- The teacher has to enter relevant solution parts and their corresponding assessment texts only once thus saving time and energy.

- He does not have to be afraid of a fast fall of his concentration since the feedback tool relieves him of the stultifying comparison of recurring patterns thus allowing him a higher degree of accuracy and objectivity in the correction.
- The time and energy saved that way can be spent more effectively for non-stereotype solutions.

Benefits for the student:

- He is urged to make greater efforts answering the exercise than with multiple choice questions, which includes a more thorough preparation.
- He receives an individual feedback upon which he cannot rest as in the case of sample solutions since he is directly required, e.g., to elaborate on a certain topic or to consult his lecturer.
- With corrected exercises he can strengthen and deepen his knowledge in a better way thus having a more thorough preparation for future examinations.

This feedback tool is still a prototype and not yet available. Future works should for example focus on the employment of a spelling control and a lexical-semantic net like GermaNet (Kunze et al., 2004). Moreover, an application to longer standardized texts could be pursued. Partial assessments within students' solutions shall be realised through quantitative methods (Schmitz, 2000; Mehler, 2004) and fuzzy logic (Rieger, 2002). A detailed description of the feedback tool is presented in Ruda (2008).

References

Austin, J. L. (1962). *How to Do Things with Words*. Harvard University Press, Cambridge, Massachusetts.

Brinker, K. (2005). *Linguistische Textanalyse. Eine Einführung in Grundbegriffe und Methoden.* 6. überarbeitete und erweiterte Auflage. Schmidt, Berlin.

Brünner, G. (2000). *Wirtschaftskommunikation. Linguistische Analyse ihrer mündlichen Formen*. Reihe Germanistische Linguistik 213: Kollegbuch. Niemeyer, Tübingen.

Heath, C. and Luff, P. (2000). *Technology in Action*. Cambridge University Press, Cambridge.

Holly, W., Kühn, P., and Püschel, U. (1984). Für einen "sinnvollen" Handlungsbegriff in der linguistischen Pragmatik. *Zeitschrift für germanistische Linguistik*, 12:275–312.

Ingenkamp, K. (1993). Erfassung und Rückmeldung des Lernerfolgs. In *Enzyklopädie Erziehungswissenschaft. Band 4: Methoden und Medien der Entwicklung des Unterrichts. 2. Auflage*, pages 173–205. Klett-Cotta, Stuttgart.

Karbach, W. and Linster, M. (1990). *Wissensakquisition für Expertensysteme. Techniken, Modelle und Softwarewerkzeuge*. Hanser, München/Wien.

Klemm, M. and Ruda, S. (2003). PortaLingua: E-Learning-Module für die Sprach- und Kommunikationswissenschaft. In Jantke, K. P., Wittig, W. S., and Herrmann, J., editors, *Von E-Learning bis E-Payment. Das Internet als sicherer Marktplatz. Tagungsband Leipziger Informatik-Tage (LIT'03), 24.–26. September 2003, Leipzig*, pages 182–191. Akademische Verlagsgesellschaft, Berlin.

Klemm, M., Ruda, S., and Holly, W. (2004). Chemnitzer E-Learning-Module für die Sprachwissenschaft. In Schmitz (2004), pages 117–131.

Kleppin, K. (2007). *Fehler und Fehlerkorrektur*, 6. Auflage. Fernstudienprojekt zur Fort- und Weiterbildung im Bereich Germanistik und Deutsch als Fremdsprache. Teilbereich Deutsch als Fremdsprache. Fernstudieneinheit 19. Langenscheidt, Berlin/München.

Kunze, C., Lemnitzer, L., and Wagner, A., editors (2004). *Anwendungen des deutschen Wortnetzes in Theorie und Praxis. Beiträge des GermaNet-Workshops. Tübingen, Oktober 2003*, volume 19(1/2) of *Journal for Language Technology and Computational Linguistics (JLCL)*.

Lienert, G. A. and Raatz, U. (1998). *Testaufbau und Testanalyse*. Beltz Psychologie VerlagsUnion, Weinheim.

Lobin, H. and Lemnitzer, L., editors (2004). *Quantitative Methoden*. Stauffenburg, Tübingen.

Mehler, A. (2004). Quantitative Methoden. In Lobin and Lemnitzer (2004), pages 83–107.

Narciss, S., Proske, A., and Körndle, H. (2004). Interaktive Aufgaben für das computergestützte Lernen. Vom ersten Entwurf bis zur technischen Realisierung. In Schmitz (2004), pages 194–206.

Neumann, O. (2003). *Wiederverwendbare Komponenten für eLearning*. PhD thesis, Technische Universität Dresden, Fakultät Informatik.

Rieger, B. (2002). Bedeutungskonstitution und semantische Granulation. In Pohl, I., editor, *Prozesse der Bedeutungskonstitution*, Sprache System und Tätigkeit 40, pages 407–444. Lang, Frankfurt a.M./Berlin.

Rothkegel, A. (1989). Textualisierung von Wissen. Einige Forschungsfragen zum Umgang mit Wissen im Rahmen computerorientierter Textproduktion. *LDV-Forum*, 6(1):3–13.

Ruda, S. (2008). *Aufgaben stellen, lösen und korrigieren. Eine sprachpragmatische Analyse für ein lehrerunterstützendes Feedback-Werkzeug im E-Learning*. Universitätsverlag Rhein-Ruhr, Duisburg.

Sächsisches Oberverwaltungsgericht Bautzen (2002). Prüfungen im Antwort-Wahl-Verfahren. Beschluss vom 10.10.2002 – 4 BS 328/02.

Schmitz, U. (2000). Statistische Methoden in der Textlinguistik. In Brinker, K., Antos, G., Heinemann, W., and Sager, S. F., editors, *Text- und Gesprächslinguistik. Ein internationales Handbuch zeitgenössischer Forschung*, Handbücher zur Sprach- und Kommunikationswissenschaft 16.1, pages 196–201. De Gruyter, Berlin/New York.

Schmitz, U., editor (2004). *Linguistik lernen im Internet. Das Lehr-/Lernportal PortaLingua*. Narr, Tübingen, (cf. `http://www.uni-due.de/portalingua/`, visited 23 March 2009).

Searle, J. R. (1969). *Speech Acts. An Essay in the Philosophy of Language*. Cambridge University Press, Cambridge.

Suchman, L. A. (1987). *Plans and Situated Actions. The Problem of Human Machine Communication*. Cambridge University Press, Cambridge.

van Someren, M. W., Barnard, Y. F., and Sandberg, J. A. C. (1994). *The Think Aloud Method. A Practical Guide to Modelling Cognitive Processes*. Academic Press, London.

von Polenz, P. (1988). *Deutsche Satzsemantik. Grundbegriffe des Zwischen-den-Zeilen-Lesens.* 2., durchgesehene Auflage. Sammlung Göschen 2226. De Gruyter, Berlin/New York.

Weber, M. (1980). *Wirtschaft und Gesellschaft. Grundriß der verstehenden Soziologie.* 5., revidierte Auflage. Mohr, Tübingen.

Winograd, T. and Flores, F. (1992). *Erkenntnis Maschinen Verstehen. Zur Neugestaltung von Computersystemen.* 2. Auflage. Rotbuch, Berlin.

Maja Bärenfänger
Angewandte Sprachwissenschaft und Computerlinguistik
Universität Gießen
Otto-Behaghel-Str. 10
D-35394 Gießen
maja.baerenfaenger@zmi.uni-giessen.de

Tobias Claas
TU Dortmund University
Martin-Schmeißer-Weg 13
D-44227 Dortmund
studiger@hytex.info

Irene Cramer
Faculty of Cultural Studies
TU Dortmund University
Martin-Schmeißer-Weg 13
D-44227 Dortmund
irene.cramer@uni-dortmund.de

Nils Diewald
Fakultät für Linguistik und Literaturwissenschaft
Universität Bielefeld
Universitätsstraße 25
D-33615 Bielefeld
nils.diewald@uni-bielefeld.de

Marc Finthammer
Information Engineering
TU Dortmund University
Martin-Schmeißer-Weg 13
D-44227 Dortmund
marc.finthammer@uni-dortmund.de

Daniela Goecke
Fakultät für Linguistik und Literaturwissenschaft
Universität Bielefeld
Postfach 10 01 31
33501 Bielefeld
daniela.goecke@uni-bielefeld.de

Mirco Hilbert
Angewandte Sprachwissenschaft und Computerlinguistik
Universität Gießen
Otto-Behaghel-Str. 10
D-35394 Gießen
Mirco.Hilbert@Germanistik.Uni-Giessen.de

Sonja Ruda
Technische Universität Chemnitz
sonja.ruda@phil.tu-chemnitz.de

Maik Stührenberg
Fakultät für Linguistik und Literaturwissenschaft
Universität Bielefeld
Universitätsstraße 25
D-33615 Bielefeld
maik.stuehrenberg@uni-bielefeld.de

Harald Lüngen
Institut für Germanistik - ASCL
Justus-Liebig-Universität Gießen
Otto-Behaghel-Straße 10
D-35394 Gießen
luengen@uni-giessen.de

Alexander Mehler
Abteilung für geisteswissenschaftliche Fachinformatik
Goethe Universität, Frankfurt am Main
Postfach: 154
Senckenberganlage 31
D-60325 Frankfurt am Main
Mehler@em.uni-frankfurt.de

Caroline Sporleder
Computational Linguistics and Phonetics
Universität des Saarlandes
Campus
D-66123 Saarbrücken
csporled@coli.uni-sb.de

Manfred Stede
Institut für Linguistik
Universität Potsdam
Karl-Liebknecht-Str. 24-25,
D-14476 Golm
stede@ling.uni-potsdam.de

Angelika Storrer
Institut für deutsche Sprache und Literatur
Universität Dortmund
D-44221 Dortmund
angelika.storrer@uni-dortmund.de