Text Parsing of a Complex Genre

Harald Lüngen, Maja Bärenfänger, Mirco Hilbert,
Henning Lobin, Csilla Puskás

Fachgebiet Angewandte Sprachwissenschaft und Computerlinguistik
Institut für Germanistik, Justus-Liebig-Universität Gießen
Otto-Behaghel-Str. 10 D, D-35394 Gießen, Germany
e-mail: {henning.lobin|maja.baerenfaenger|mirco.hilbert|
harald.luengen|csilla.puskas}@uni-giessen.de

Abstract

A text parsing component designed to be part of a system that assists students in academic reading an writing is presented. The parser can automatically add a relational discourse structure annotation to a scientific article that a user wants to explore. The discourse structure employed is defined in an XML format and is based the Rhetorical Structure Theory. The architecture of the parser comprises pre-processing components which provide an input text with XML annotations on different linguistic and structural layers. In the first version these are syntactic tagging, lexical discourse marker tagging, logical document structure, and segmentation into elementary discourse segments. The algorithm is based on the shift-reduce parser by Marcu (2000) and is controlled by reduce operations that are constrained by linguistic conditions derived from an XML-encoded discourse marker lexicon. The constraints are formulated over multiple annotation layers of the same text.

Keywords: text parsing; discourse parsing; XML applications; rhetorical structure

1 Introduction and Motivation

Text Parsing deals with the automatic allocation of structure to whole texts, in analogy to sentence parsing which deals with the allocation of structure to single sentences. More specifically we talk about *discourse parsing* when the target structure and discourse interpretation strategies conform to a *discourse theory* such as the Rhetorical Structure Theory (RST, [[9],[10]]). Rhetorical structures as devised in RST provide an analysis of an input text in terms of splitting it into discourse segments ordered in a hierarchy, and of specifying functional-argumentative relations between them, such as *Cause, Background*, or *Elaboration.*

Discourse parsing has traditionally dealt with shortish text types such as newspaper articles or encyclopaedia entries [[6],[10],[17]], and has been based on the discourse features of lexical discourse markers (connectives) and a syntactic analysis of the input text. In our approach, we also regard connectives and syntax as the major source of discourse interpretation on the micro level of discourse, i.e. of discourse segments that are smaller than paragraphs. But to parse the macro level of relational discourse structure of instances of a more complex text type, we consider information on other levels as (more abstract) discourse markers, too, such as the logical document structure (division into sections, subsections, paragraphs etc.), thematic structure (e.g. anaphoric structure and lexical chains), and genre-specific text-type structure. To make the different sources of discourse marking features compatible and available for a discourse parser, we devised an architecture where information on different linguistic and text structural levels is stored in separate XML annotation layers of the input text.
What is the benefit of a parser that provides a text, for example a scientific article, with a rhetorical structure? Since in a rhetorical structure text segments are classified into nuclei (nuclear propositions of the text that are essential to the author's intentions) and satellites (segments that provide only additional information and that can be potentially omitted), discourse parsing can be a major step in automatic text summarisation, cf. [[6],[10]], and [[15]]. The application that we have in mind in our project is a different one. We designed a project scenario where a system supports students in developing adequate strategies for reading scientific articles. The system shall have two dimensions: It shall provide a tool which supports students in explorative reading, and it shall function as a learning environment where students can learn something about the characteristics of the genre of "scientific article", its generic text type structure (assignment of text segments to text-type specific functional categories such as *introduction, method, results, discussion*), its rhetorical structure (including argumentative strategies), and the unfolding of thematic progression. The support of explorative and selective reading is based on two mechanisms: highlighting text structure and providing – automatically generated – link lists to different structural elements as navigation elements. By following links or by directing the attention on highlighted

passages, readers are guided to thematically or rhetorically interesting parts of the text, and they are also supported in selective reading and cross-reading. Additionally, the access to different structural levels of the text is simplified, because its building plan is made explicit.

Highlighting and linking requires the pre-processing of articles. They must automatically be analysed and annotated on the levels of document structure, generic text type structure, rhetorical and thematic structure. The automatisation of analysis and annotation is necessary to enable users of the system to upload articles that they consider relevant themselves. To allow students such a personalised use of the system, a discourse parser is being developed which implements the task of automatically adding discourse structure annotations.

2 Discourse Analysis

Three current theories of discourse are the *Unified Linguistic Discourse Model* (ULDM, cf. [[14],[15]]), the *Segmented Discourse Representation Theory* (SDRT, cf. [[1],[2]]), and the *Rhetorical Structure Theory* [[9],[10]]. All three theories have been used in the automatic analysis of discourse previously. They share the following assumptions, which are illustrated in the RST discourse tree shown in Figure 1 which models a passage out of [[3]]:

1. Discourse relations hold between *discourse segments*. Discourse segments can be *elementary* or *complex.* In Figure 1, each horizontal line spans one discourse segment. Thus, in the structure there are three elementary discourse segments (EDS, i.e. those that directly span text), and two complex discourse segments (CDS, those that directly span two smaller, elementary or complex, discourse segments.
2. Discourse is structured hierarchically. The overall structure of discourse corresponds to a graph structure (as in SDRT) or even to a tree structure (as in ULDM and RST).
3. There are two major structural types (a.k.a. schemas) of discourse relations: hypotactic (or mononuclear, or subordinating) relations, and paratactic (or multinuclear, or coordinating) relations. In Figure 1, the *Elaboration* relation is hypotactic, i.e. one of its elements has the status of being the nucleus, “the more salient or essential piece of information” [[5]] of the relation. The other relation, *Sequence*, is paratactic, with two nuclei, i.e. elements of equal salience.

According to [[20]], discourse parsing can be viewed as consisting of three subtasks which in our formulation are the following:

1. The segmentation of the input text into elementary discourse segments (EDS)
2. The identification of segments that can be recursively combined into larger (complex) discourse Segments (CDS)
3. The assignment of discourse relations between the segments that belong to one CDS

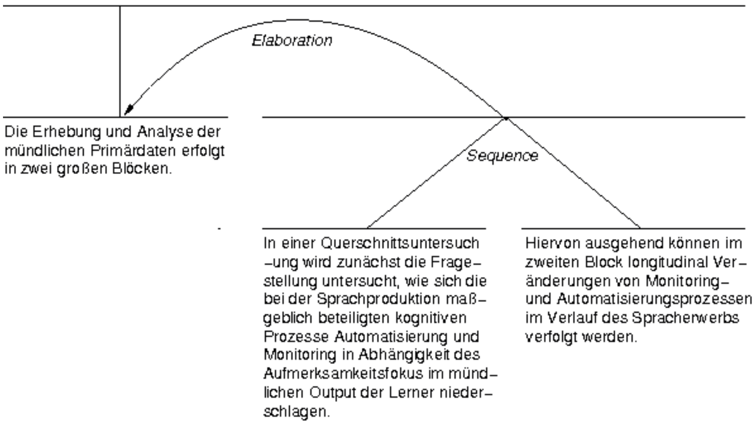
Elaboration
Die Erhebung und Analyse der mündlichen Primärdaten erfolgt in zwei großen Blöcken.
Sequence
In einer Querschnittsuntersuch-ung wird zunächst die Fragestellung untersucht, wie sich die bei der Sprachproduktion maßgeblich beteiligten kognitiven Prozesse Automatisierung und Monitoring in Abhängigkeit des Aufmerksamkeitsfokus im mündlichen Output der Lerner niederschlagen.
Hiervon ausgehend können im zweiten Block longitudinal Veränderungen von Monitoring- und Automatisierungsprozessen im Verlauf des Spracherwerbs verfolgt werden.

Figure 1: RST tree of a short passage of [[3]]

2.1 Discourse Markers

Discourse markers are elements that signal the discourse (rhetorical) relation between two text segments. We distinguish different types of discourse markers. Firstly, there are lexical discourse markers, or connectives. These are syntactically mostly adverbs or conjunctions. The German subordinating conjunction *obwohl*, for example, indicates a relation of *Concession* between its containing elementary discourse segment and the adjacent segment that contains its main clause, either preceding or following. Secondly, grammatical or document type-related features can function as discourse markers. The occurrence of an <itemizedList>-environment in the logical document structure, for example, indicates a paratactic *ListSequence* relation. For the first type of markers (the lexical ones), we have introduced an XML discourse marker lexicon format. A discourse marker entry consists of an identification unit, a filter unit, and an allocation unit. In the identification unit, a discourse marker is identified by its textual representation plus its lemma plus its POS tag. The optional filter unit includes hypotheses about possible contexts that disambiguate a discourse markers with respect to its associated rhetorical relation. The allocation unit contains optional conditions for the relations induced by the discourse marker. The second type of discourse marker is so far only introduced in the hypothesis section of a lexical discourse marker.

```
<dm id="c19" typ="lexical">
  <cue>
    <text>:</text>
    <lemma pos="PUNCT">:</lemma>
    <position><end>+</end></position>
  </cue>
  <kommentar>wenn hinter "glossterm", dann ELA-DEF in nuk</kommentar>
  <filter>
    <hypothese relname="Elaboration-definition">
       <doc>self::doc_glossterm</doc>
    </hypothese>
    <hypothese relname="Elaboration-derivation">
      <seg>following-sibling::eds</seg>
                                 </hypothese>
  </filter>
  <rels default="Preparation-other">
    <relation relname="Preparation-other" skopus="sds+" typ="s" beds-
      richtung="r"></relation>
    <relation relname="Elaboration-definition" skopus="sds+" typ="n" beds-
      richtung="r"></relation>
    <relation relname="Elaboration-derivation" skopus="sds" typ="n" beds-
     richtung="r"></relation>
  </rels>
</dm>
```

Listing 0: Discourse marker lexicon entry for a colon

In Listing 0, which shows the entry for the colon (':') as a lexical discourse marker, the different sections of an entry are illustrated. The entry is uniquely identified by the content of the <cue> element, i.e. its text, lemma, part-of-speech, and position information. The filter unit indicates two readings (<hypothese>) of the colon: If it is contained in a <glossterm> element on the logical document structure layer, it is involved in an *Elaboration-definition* relation. If its following sibling element on the segmentation layer is of type <eds>, it induces an *Elaboration-derivation* relation. In the allocation unit marked by <rels>, the default relation that is the target when none of the conditions in the filter unit are fulfilled is given. Moreover, for each of the three alternative relation assignments (<relation>), its scope (@skopus) is indicated, e.g. 'sds+' means that the colon in this reading may combine only with units above sentence level. Furthermore, the nuclearity type of the containing segment is indicated in the @typ attribute ('n' or 's' for satellite or nucleus), as well as the direction of the segment it is to be combined with in the @beds-richtung attribute ('l' or 'r' for left or right).

Each entry in the discourse marker lexicon gives rise to one ore more reduce rules used in the parser, cf. Section 3.2.1. The discourse lexicon is compiled from a list of connectives extracted from our development corpus of 47 German scientific articles from the journal *Linguistik Online*. We work with a relation set of altogether 42 base relations which are grouped into relation classes. The relation set was defined considering several relation sets from previous projects ([[5],[8],[9]]), and examining their relevance with respect to the RST annotations of our corpus and the scenario.

2.2 Discourse Structure Representation

We represent the target rhetorical structures in XML such that the XML document tree mirrors the hierarchical structure of discourse, i.e. the subnode relation in a discourse tree is always represented by XML element containment. Thus our format is distinguished from two previous approaches to XML-based representation of RST structures, firstly, the one that is output by O'Donnell's RST annotation tool [[13]], and secondly, the Underspecified Rhetorical Markup Language URML as put forward in [[18]]. In the former, all non-empty elements correspond to elementary discourse segments, while span, nuclearity, and relational information about complex discourse segments is stored in empty elements with IDREF attributes. The latter, URML, is so far the most comprehensive account of an XML-based markup language for the representation of RST structures. It provides techniques for representing underspecified analyses on several levels, e.g. relation types and nuclearity types, as well as types of structural ambiguity using ID references. Underspecification in URML is employed for representing uncertain or ambiguous analyses in manual annotation but also for storing concurrent analyses which have not (yet) been disambiguated in a discourse parsing process.

```
<para xsi:noNamespaceSchemaLocation="hypo-para.v4.0.xsd" relname="contrast"
 id="i1">
  <n id="i2">
    <hypo id="i3" relname="elaboration-example">
      <n id="i4">
        <t id="i5">In der Schrift hat die Sprachpflege einen etwas besseren
           Erfolg als im Gespräch gehabt.</t>
      </n>
      <s id="i6">
        <t id="i7">In öffentlichen Dokumenten ist man z.B. darauf bedacht,
           dass die Termini dem Gebrauch in Schweden entsprechen.</t>
      </s>
    </hypo>
  </n>
  <n id="i8">
    <t id="i9">Trotzdem enthalten sowohl Sachtexte als auch die Belletristik
       sprachliche Züge, die den Schweden fremd vorkommen.</t>
  </n>
</para>
```

Listing 0: RST-HP Format

Contrary to the above mentioned approaches, we designed an XML application of an RST format that we call RST-HP (after the two central elements, <hypo> and <para>). The format is an extension of one developed in a previous project [[12]]. An example of an RST tree in RST-HP is given in Listing 0 (the text is an extract from [[3]]). The XML content model for <hypo> comprises exactly one <n> and one <s> in any order. The content models for <n> and <s> are identical, comprising either one <hypo>, one <para>, or one <t>, the latter being a terminal, i.e. an element with textual content. The actual rhetorical relation label is contained in the @relname attribute at the <para> and <hypo> elements. The proposed structure comes close to what is called the dependency tree representation in [[7]], the difference being that, in dependency trees, N and S are arc labels rather than intermediate nodes. For the representation of alternative analyses of (parts of) the same text we employ XML-based multi-layer annotation [256].

We believe that the RST-HP format is much more in the spirit of a "text-technological modelling of information" than a format that relies on IDs/IDREFs to encode the discourse tree. Moreover, the retrieval of subtrees of an

```
<embed xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:noNamespaceSchemaLocation="hypo-para.v4.0.xsd" id="i21"
 relname="elaboration-specification-other">
  <n id="i22">
    <t id="ti35">Monitoring ist ein spezieller Aufmerksamkeitsprozess, der bei
      jeglicher sprachlicher Performanz
       <s id="i23">
         <t id="ti36"> - sowohl in der L1 als auch in der L2 - </t>
       </s>
     zu beobachten ist.</t>
  </n>
</embed>
```

Listing 0: Embedded satellite in RST-HP

RST structure of a text seems much easier in our model, using e.g. XPATH expressions or the Prolog tool set described in [256]. It is not necessary to permit unlimited graph structures for those aspects of discourse structure that cannot be described by trees. For adding graph properties to the model, ID references can be used and restricted to those cases where they are required. Since the ID/IDREF mechanism has a secondary status in the XML model, we give such structures a secondary, exceptional status in the interpretation of the model, too. We use them for example to encode dislocated satellites (cf. [[16]]), i.e. satellites that are not adjacent to their nuclei. In our corpus of scientific articles, they correspond mostly to (potential) "floating" objects like, tables, images, and footnotes, violating the adjacency constraint of RST originally formulated in [[9]]. A dislocated satellite is accommodated by allowing an <extra> element somewhere interspersed with <n> or <s> elements. Its @id has to be "bound" by an @extraref attribute of an <s> elsewhere in the tree, which is either empty or contains only a referring expression. Another construction where the @extraref attribute is used involves cases where one discourse segment seems to be an argument of more than one discourse relation. In scientific articles this sometimes occurs with two consecutive list constructions, where the items of the second list seem to elaborate on the items of the first list.

Finally, our format provides an additional <embed> element construction for segments that disrupt other segments in the linear text, e.g. with dashes as discourse markers. We found that in general, such embedded segments are related to their embedding segments like a satellite in a hypotactic construction, thus the content of an <embed> is defined as an <s> which is contained in the text() of a <t>, which is the node indicating a terminal under <n> (Listing 0).

3 System

3.1 Architecture

Our approach to discourse parsing includes auxiliary analysis components that provide the parser with features of the document on different levels, encoded in XML annotation layers. Each annotation layer also contains the input text so that the input text can be used as a connection between the layers in the line of XML-based multi-layer annotation [[24]]. Other resources, such as a taxonomy of rhetorical relations and a discourse marker lexicon, as well as the target structure are represented in XML, too. Text parsing is thus the addition of another, higher level annotation layer.

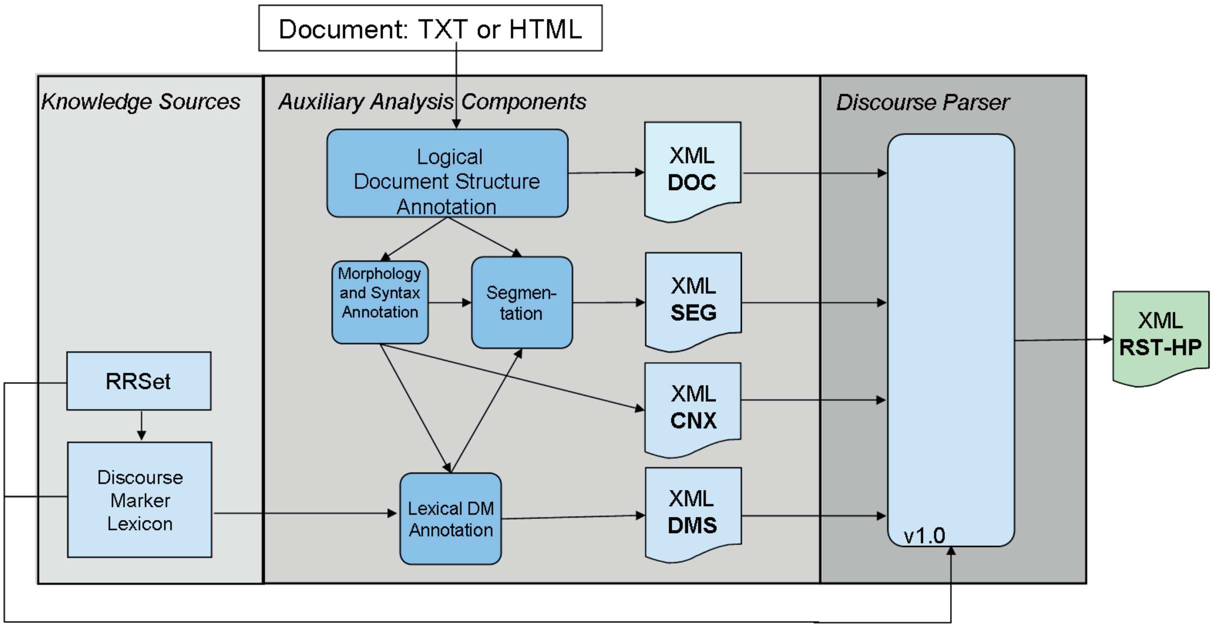

Figure 1: Architecture of the discourse parsing system

Version v1.0 of the parser is based on two knowledge source (the rhetorical relation set RRSet and the discourse marker lexicon DMList, both to be described below) and on the following four input layers, cf. Figure 1:

RST structure of a text seems much easier in our model, using e.g. XPATH expressions or the Prolog tool set described in [256]. It is not necessary to permit unlimited graph structures for those aspects of discourse structure that cannot be described by trees. For adding graph properties to the model, ID references can be used and restricted to those cases where they are required. Since the ID/IDREF mechanism has a secondary status in the XML model, we give such structures a secondary, exceptional status in the interpretation of the model, too. We use them for example to encode dislocated satellites (cf. [[16]]), i.e. satellites that are not adjacent to their nuclei. In our corpus of scientific articles, they correspond mostly to (potential) “floating” objects like, tables, images, and footnotes, violating the adjacency constraint of RST originally formulated in [[9]]. A dislocated satellite is accommodated by allowing an <extra> element somewhere interspersed with <n> or <s> elements. Its @id has to be “bound” by an @extraref attribute of an <s> elsewhere in the tree, which is either empty or contains only a referring expression. Another construction where the @extraref attribute is used involves cases where one discourse segment seems to be an argument of more than one discourse relation. In scientific articles this sometimes occurs with two consecutive list constructions, where the items of the second list seem to elaborate on the items of the first list.

Finally, our format provides an additional <embed> element construction for segments that disrupt other segments in the linear text, e.g. with dashes as discourse markers. We found that in general, such embedded segments are related to their embedding segments like a satellite in a hypotactic construction, thus the content of an <embed> is defined as an <s> which is contained in the text() of a <t>, which is the node indicating a terminal under <n> (Listing 0).

3 System

3.1 Architecture

Our approach to discourse parsing includes auxiliary analysis components that provide the parser with features of the document on different levels, encoded in XML annotation layers. Each annotation layer also contains the input text so that the input text can be used as a connection between the layers in the line of XML-based multi-layer annotation [[24]]. Other resources, such as a taxonomy of rhetorical relations and a discourse marker lexicon, as well as the target structure are represented in XML, too. Text parsing is thus the addition of another, higher level annotation layer.

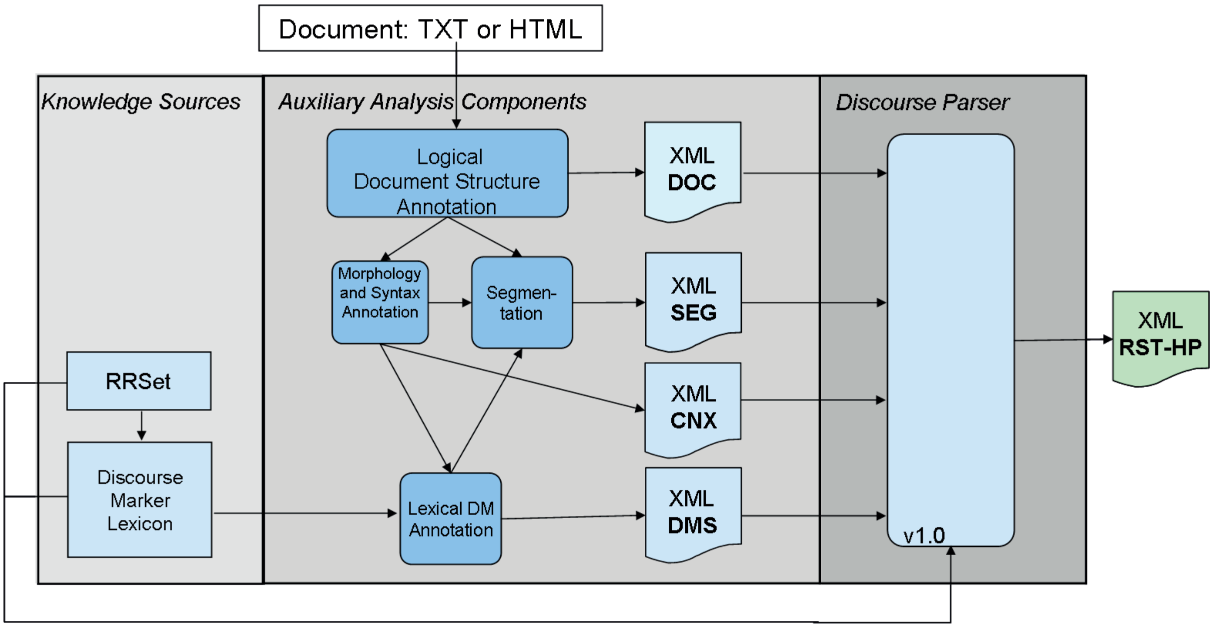

Figure 1: Architecture of the discourse parsing system

Version v1.0 of the parser is based on two knowledge source (the rhetorical relation set RRSet and the discourse marker lexicon DMList, both to be described below) and on the following four input layers, cf. Figure 1:

- CNX: Syntax and morphology layer, provided by the robust syntactic parser-tagger *Machinese Syntax* by Connexor Oy. Machinese is based on Functional Dependency Grammar [[21]] and can be parametrised to put out XML. We have reconverted the output slightly to (a) preserve primary data identity with the original text and (b) to make certain tags more explicit, e.g. the POS tag which is originally "hidden" in a string.
- DOC: Logical document structure layer based on DocBook [[23]]. The DOC layer is so far added by running conversion scripts on HTML or plain text generated from PDF, the outputs of which have to be manually corrected. A better automatisation of this pre-processing step will be implemented at a later stage of the project.
- SEG: Initial segmentation layer. Based on the logical document structures and syntax/morphology, the input text is segmented into clause-like elementary discourse segments (EDS). In defining EDSs we have closely followed the definition of EDUs in [[5]], but adapted the punctuational and grammatical criteria to German and also to our text type and application scenario. Relative clauses, for example, are not considered separate EDSs, because restricting relative clauses are not independent discourse segments but modify the meaning of the head N, and they cannot be distinguished from non-restricting relative clauses by syntax and punctuation in German, either. The segmentation is provided by a Perl program that reads in the text and DOC annotation layer and repeatedly calls the CNX parser during execution. The program also tags two other kinds of discourse segments: SDS (discourse segments corresponding to complete sentences) and CDS (complex discourse segments identified by paragraphs and higher structural elements on the DOC layer). The discourse parser is not allowed to construct discourse segments that are incompatible with the SDSs and CDSs found on the SEG layer.
- DMS: A layer including the annotation of potential lexical discourse markers. When the CNX annotation of an input word matches the text, lemma, and POS specifications of an entry in the discourse marker lexicon, the input word is tagged as lexical discourse marker. All subsequent disambiguation and relation assignment is done by the discourse parser. Only in the case of the conjunction *und* ('and'), it is checked whether it is S-coordinating or VP-coordinating; otherwise it is not tagged.

Our development corpus of 47 German linguistic articles is annotated on all levels involved, plus also a (genre-specific) text type structure layer (see [[4]]) (not used in the present implementation). In later phases of the project, further layers will be input to the parser, such as the text-type structure and thematic structure.

3.2 Parser

This section describes the algorithm of the discourse parser based on the resources described so far. Version v1.0 as shown on the right hand side of Figure 1 will be further described. Besides the incoming text as such (a scientific article), it has four annotation layers of the text as its input: its logical document structure (DOC), its initial segmentation (SEG), syntax and morphology tagging (CNX), and the lexical discourse marker annotation (DMS).

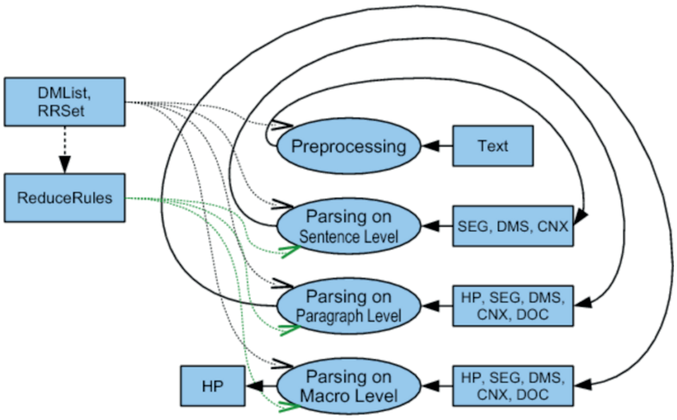

Figure 2: Parsing in loops

The parser uses a bootstrapping technique in iteratively generating revised or new XML annotations based on the current input annotation layers. The layer SEG is used to guide the bottom up reductions. In the first loop, discourse relations are hypothesised to build segments corresponding to SDS elements (sentential discourse segments) on the SEG layer. In the second loop, these segments are iteratively combined to form segments corresponding to CDS (complex discourse segments) on the paragraph level. In the next loop, these in turn are combined to CDS on the section level; and so forth, until the level of the whole document is reached. In each of these loops, a different set of features (i.e. annotation layers) needs to be considered. The layers corresponding to the thematic structure or text type structure, for example, will only be considered above paragraph level.

The final output is a set of possible annotations of the discourse structure in the extended HP format described in the previous section. The most highly ranked discourse structure can then serve as an input to the explorative reading scenario introduced in Section 1.

For the parsing algorithm, we follow a similar approach as one of those proposed in [256], namely a shift-reduce parser controlled by reduce operation rules that are derived from linguistic constraints. These constraints describe correlations between different discourse markers and configurations on the different annotation layers and yield hypotheses of rhetorical relations holding between elementary and complex discourse segments of the input text.

After loading and preparing the input resources and generating the reduce operation rule set, the input annotations are parsed using the shift-reduce parser.

3.2.1 *Reduce operation rules*

To generate the reduce operation rule set, abstract rule templates are employed which correspond to the different kinds of rhetorical schemata to be built, i.e. schemata for hypotactic, paratactic, or embedded constructions as described in Section 2.2. These rule templates are filled with information from the knowledge sources (mainly constraints from the discourse marker lexicon) to specify a hypothetical rhetorical tree. Listing 0 shows a non-terminal reduce rule, which introduces the hypotactic relation *Concession* between two adjacent discourse segments.

```
<reduce id="r2" source="hlu">
  <in>
    <hpx:undefined id="$idN">$contentN</hpx:undefined>
    <hpx:undefined id="$idS">$contentS</hpx:undefined>
  </in>
  <out>
    <hpx:undefined id="generate-id()">
      <hpx:hypo relname="Concession">
        <hpx:n id="$idN">$contentN</hpx:n>
        <hpx:s id="$idS">$contentS</hpx:s>
      </hpx:hypo>
    </hpx:undefined>
  </out>
  <constraint test="text-inclusion($contentS, dm:dm[.='obwohl'])"/>
  <constraint test="same-sds($idN, $idS)"/>
  <constraint test="identity($contentS, seg:eds"/>
  <constraint test="identity($contentN, seg:eds"/>
</reduce>
```

Listing 0: Non-terminal reduce rule introducing a hypotactic relation containing one nucleus and one satellite element

By the operation described in the rule, the status of each segment is changed from <undefined> to <n> (nucleus) or <s> (satellite, respectively). Each reduce operation introduces a new containing <undefined> element, which is to be replaced by <n> or <s> in later parsing steps. An <undefined> element represents a node in a discourse tree with an undefined nuclearity status, modelled after the node label 'status=undefined' used in [10].

Listing 0 shows another kind of reduce rule. Here the relation *Elaboration-specification-other* is introduced to hold between a nucleus and its embedded satellite. The relation is marked by the dashes enclosing the content of the second input segment in combination with the absence of a main verb tag on the CNX layer of the same input segment.

When applying the rule in one reduce operation, the content of the <in> element is replaced by the content of the <out> element. In doing so, variables like $id or $content transfer their values from the <in> to the <out> element, and functions like generate-id() are executed. The applicability of a rule is restricted by Boolean conditions expressed by one or more <constraint> elements.

The set of functions used here in attribute values are adopted XPATH functions like generate-id() or additional internally defined functions like contains() and dm(). The internal functions are defined according to the schema grammars of the knowledge sources and are realised by basic XML queries and operations using XPATH, or different API functions, e.g. from the LibXML library. Access to the different input resources is handled using namespaces combined with resource-specific access functions like dm(). That way, lexical and syntactic information can be combined with information about the logical structure of the content of the input segments and/or other kinds of discourse markers.

```
<reduce id="r3" source="mhi">
  <in>
    <hpx:undefined id="$idN1"><hp:t id="$idT1">$contentT1</hp:t></hpx:undefined>
    <hpx:undefined id="$idS">$contentS</hpx:undefined>
    <hpx:undefined id="$idN2"><hp:t id="$idT2">$contentT2</hp:t></hpx:undefined>
  </in>
  <out>
    <hpx:undefined id="generate-id()">
      <hp:embed relname="Elaboration-specification-other">
        <hp:n id="$idN1">
          <hp:t id="$idT1">$contentT1
            <hp:s id="$idS">$contentS</hp:s>
          $contentT2</hp:t>
        </hp:n>
      </hp:embed>
    </hpx:undefined>
  </out>
  <constraint test="surrounded-by($contentS, dm(' - '), dm(' - '))"/>
  <constraint
       test="not-contains($contentS, cnx:/sentence/token/depend/@value='main'")/>
</reduce>
```

Listing 0: Non-terminal reduce rule introducing a hypotactic relation containing a nucleus with an embedded satellite element

3.2.2 Parsing the input annotations

The input annotation layers and further knowledge sources are loaded as XML tree objects. In XML-based multi-layer annotation, the primary textual data are annotated several times according to each annotation layer. However, also for efficiency reasons, the internal representation of the parser rather corresponds to a standoff representation ([22]). That is, the primary data in all input annotations are replaced by primary data references.

The shift-reduce parser takes the SEG annotations as its main input. Each EDS is treated as a terminal unit. The processing strategy of the parser is bottom-up, left-to-right, and depth-first.

Thus, after shifting an elementary discourse segment from the SEG annotation to the parsing stack, the parser tries to reduce the top of the stack as far as possible. First it applies a terminal reduce rule introducing a new <hp:t> element. Then it tries to apply non-terminal rules matching the top of the stack (cf. [[10]], pp. 152). If more than one rule is applicable, the possible alternatives will be regarded as hypotheses and pursued in parallel. When parsing an average article with about 500 elementary discourse segments, different heuristics are used to combine the emitted hypothesis during parsing:

- If two (or more) different relations are proposed to hold between two segments, the taxonomy of relations defined in the RRSet can be consulted to find a common parent relation.
- The reduce rules can be augmented with weights which can be estimated by inductive learning, where the automatic annotation of a set of training articles is compared with their manual annotations. The weight of the reduce rule is then increased when it agrees with the manual annotation regarding the discourse markers and relation assignment. To this end, each reduce rule is provided with an @id attribute, and each introduced <hypo>, <para>, and <embed> element will have a reference to the reduce rule.

4 Conclusions

We presented a text parsing component designed to be part of a system that supports students in selective and efficient reading of scientific articles and in acquiring knowledge about academic writing. In this scenario, the text parser can automatically add a discourse structure annotation to an article that a user wishes to explore. The discourse structure we employ is based on Rhetorical Structure Theory (RST). We presented our text-technological architecture for discourse parsing, in which several pre-processing components provide an input text with linguistic annotations in different XML annotation layers.

We the introduced resources and methods based on XML technologies that we employ in developing the parser. First, the structure of a discourse marker lexicon was sketched which in version v1.0 contains the lexical discourse markers. Its entries provide information used for a disambiguation of markers with respect to the rhetorical relations they signal. Moreover, the target format representing the discourse structure that is supposed to be the output of the parser was presented. It employs the XML document tree to model discourse trees according to RST. Additional XML constructions were introduced to represent deviations from the RST-tree structure, which are still relevant in to our corpus. Finally, the discourse parser proper and its processing strategies were discussed. Its algorithm is based on the shift-reduce parser for discourse parsing introduced in [[10]]. It is controlled by reduce operations that are derived from linguistic constraints which in version v1.0 are mainly generated from the discourse marker lexicon. There are also reduce rules that process constraints on the levels of logical document structure and syntax/morphology, such as the ones generated from the colon-entry in the discourse marker lexicon.

The project is presently in its first of three years, i.e. the resources are built, and the parser is being implemented. It will be internally evaluated on a test set of articles from our corpus, for which we have created manual discourse structure annotations to serve as a standard. An external evaluation will be arranged with one of our project partners in the context of automatic hypertextualisation.

Besides increasing the coverage of the existing resources, the main concerns in the future will firstly be on handling discourse relations that are not signalled by lexical discourse markers. This is increasingly important for the more global level of discourse, i.e. in interrelating paragraphs and higher level segments. Of specific interest is the *Elaboration* relation, which in our opinion is used to encode aspects of thematic progression in RST. For an identification of *Elaboration* and its several subtypes, we intend to explore an approach using lexical cohesion [[11]] and anaphoric structure. Lexical cohesion also plays an important role in determining complex discourse segments besides the ones that are given in the logical document structure, and to detect mismatches between the discourse structure and the logical document structure. Finally we will focus on an automatic detection of text areas that are instances of categories from the genre-specific text type structure and on how such structures can be combined with relational discourse structure.

References

[1] ASHER, N.; LASCARIDES, A. *Logics of conversation*., Cambridge U.K. : Cambridge University Press, 2003.

[2] ASHER, Nicholas; VIEU, L. Subordinating and coordinating discourse relations. *Lingua,* 2005, vol. 115, no. 4, p. 591-61.

[3] BÄRENFÄNGER, O.; BEYER, S. Zur Funktion der mündlichen L2-Produktion und zu den damit verbundenen kognitiven Prozessen für den Erwerb der fremdsprachlichen Sprechfertigkeit. *Linguistik Online*, 2001, no. 8. URL http://www.linguistik-online.de, 28.2.2006.

[4] BAYERL, P.S.; LÜNGEN, H.; GOECKE, D.; WITT, A.; NABER, D. Methods for the semantic analysis of document markup. **In** *Proceedings of the ACM Symposium on Document Engineering (DocEng 2003).* Grenoble, 2003, p. 161-170.

[5] CARLSON, L; MARCU, D. Discourse tagging reference manual. *Technical Report ISI-TR-545*. Marina del Rey CA: Information Science Institute. 2001.

[6] CORSTON-OLIVER, S. (1997). *Computing of representations of the structure of written discourse.* Ph.D. thesis. Santa Barbara: University of California, 1997.

[7] DANLOS, L. Comparing RST and SDRT discourse structures through dependency graphs. **In** *Proceedings of Constraints in Discourse.* Edit. by C. Sassen, and A. Benz and P. Kühnlein. Dortmund, Germany, 2005, p. 46-53.

[8] HOVY, E.; MAIER, E. Parsimonious or profligate: How many and which discourse structure relations? 1995. Unpublished ms. URL http://www.isi.edu/natural-language/people/hovy/publications.html, 28.2.2006.

[9] MANN, W. C.; THOMPSON, S. A. Rhetorical Structure Theory: Toward a functional theory of text organisation. *Text,* 1988, vol. 8, no. 3, p. 243-281.

[10] MARCU, D. The theory and practice of discourse parsing and summarization. Cambridge MA : MIT Press, 2000.

[11] MORRIS, J.; HIRST, G.Lexical cohesion, the thesaurus, and the structure of text. *Computational linguistics*, 1991, vol. 17, no. 1, 21-48.

[12] MINNING, H.; PUSKAS, C.; SALISBURY, J. Rhetorisches Parsing deutschsprachiger Texte. **In** *Proceedings of the 12th Student Conference on Computational Linguistics (TaCoS 2002).* Potsdam, Germany, 2002.

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

[14] POLANY, L.; CULY, C.; van den BERG, M.; THIONE, G. L.; AHN, D. A rule-based approach to discourse parsing. In *Proceedings of the 5th Workshop in Discourse and Dialogue,* Cambridge MA, 2004, p. 108-117,.

[15] POLANY, L.; CULY, C.; van den BERG, M; THIONE, G.L.; AHN, D. Sentential structure and discourse parsing. In *Proceeedings of the ACL 2004 Workshop on Discourse Annotation*. Barcelona, 2004, p. 49-56,.

[16] POWER, R.; SCOTT, D.; BOUAYAD-AGHA, N. Document structure. *Computational Linguistics,* 2003, vol. 29, no. 2, 211-260.

[17] REITTER, D. Simple signals for complex rhetorics: On rhetorical analysis with rich-feature support vector models. **In** *Sprachtechnologie für die multilinguale Kommunikation. Textproduktion, Recherche, Übersetzung, Lokalisierung.* Edit. by Uta Seewald-Heeg. *Beiträge der GLDV-Frühjahrstagung*, Köthen, Germany, 2003, vol. 8 of LDV-Forum, p. 38-52.

[18] REITTEER, D.; STEDE, M. 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, 2003.

[19] SAARI, M. Schwedisch als die zweite Nationalsprache Finnlands: Soziolinguistische Aspekte. Linguistik Online.. 2001, no. 8. URL http://www.linguistik-online.de, 28.2.2006.

[20] SPORLEDER, C.;. LASCARIDES, A. Combining hierarchical clustering and machine learning to predict high-level discourse structure. In *Proceedings of COLING-04.* Geneva, Switzerland, 2004, p. 43-49.

[21] TAPANAINEN, P.; JÄRVINEN, T. A non-projective dependency parser. In *Proceedings of the 5th Conference on Applied Natural Language Processing.,* Washington D.C., 1997, p. 64-71.

[22] THOMPSON, H. S.; McKELVIE, D. Hyperlink semantics for standoff markup of read-only documents. **In** *Proceedings of SGML Europe 1997: The next decade – Pushing the Envelope*, Barcelona. 1997.

[23] WALSH, N.; MUELLNER, L. *DocBook: The Definitive Guide,* O'Reilly. 1999.

[24] WITT, A. Multiple hierarchies: New aspects of an old solution. **In** *Proceedings of the Extreme Markup Languages*. Montreal. 2004.

[25] WITT, A.; LÜNGEN, H.; GOECKE, D.; SASAKI, F. Unification of XML documents with concurrent markup. *Literary and Linguistic Computing,* 2005, vol. 20, no. 1, p. 103-116.