Section Structure (`Struct-1`): We noticed that apart from global locational structure, there is also a section internal locational organization which might be important for Argumentative Zoning. Introductions usually proceed from the more general to the more specific, with general knowledge typically coming first and statements about own work appearing towards the end. In particular, AIM sentences often occur in a typical position about two-thirds down in introduction sessions.

We also observed that the first and last sentences in other sections often fulfill a summarizing function, and are often associated with text-organization meta-discourse (*"in this section we will"*), which is captured by our TEXTUAL sentences. The second and third or second and third-last sentence also often have a special summarizing function.

The feature `Struct-1` divides the section into three equally sized segments, and additionally singles out the first and the last sentence, and takes together the second and third sentence as a sixth value, and the second-last plus third-last sentence as a seventh value.

Paragraph Structure (`Struct-2`): There is disagreement in the literature whether paragraph information should be considered as a surface indicator of importance and topic boundaries. Are paragraphs regarded as logical units by authors, or rather as layout units?

(Baxendale, 1958) states that due to the hierarchical organization of well-written research papers, sentences at the beginning and end of the paragraph are more likely to be "topic sentences"—in 85% of the paragraphs, the topic sentence was the initial sentence, and in 7% the final. Marcu (1997b) also suggests that paragraph breaks help readers determine the most important textual units in a text.

In contrast, Longacre (1979) holds that the function of many paragraph breaks is purely aesthetic, and Starck (1988) conducted an experiment which confirms the marginal role of paragraphs in higher-level interpretive tasks. The task of human re-introduction of paragraph breaks led to poor results: only nine of the 17 paragraph breaks in a text were correctly identified as such by more than 50% of the subjects. We lean towards the layout argument: we believe that in conference papers, the number and placement of paragraph breaks will be affected by the question whether or not a paper was printed in "two-column" style.

Even if we do find crucial information at the beginning and the end of paragraphs, we still do not know how useful this is for Argumentative Zoning. With respect to other tasks, Hearst (1997) indicates that thematic boundaries do not always occur at paragraph boundaries, but Wiebe (1994) states that the information whether or not a sentence begins a paragraph is useful for her task, namely the determination of private-state sentences in narrative (subjective vs. objective orientation). In our case, it seems sensible to assume that CONTRAST sentences are more likely to occur at the end of a paragraph, but other than that it seems difficult to predict a direct correlation between paragraph boundaries and argumentative flow. We included the feature in our heuristics pool to determine its usefulness empirically.

Headlines (`Struct-3`): Van Dijk (1980) states that in scientific articles, rhetorical sections are marked by fixed headlines. Knowing which rhetorical section a sentence belongs to should be directly useful for Argumentative Zoning. For example, Nanba and Okumura (1999) assume a correlation between rhetorical section and type of citation. They expect CONTRAST citations to occur more often in the sections *Introduction, Discussion*, and *Related* work, and BASIS citations to occur more often in the *Introduction* and the *Method* section.

However, we have argued in section 3.1 that not all articles in our corpus keep to a fixed section structure. As a result, we expect the feature `Struct-3` to be of use only in those cases where prototypical headings are available.

Feature `Struct-3` classifies the headlines into groupings of similarity on semantic grounds and morphological variants, resulting in the following 15 classes: *Introduction, Problem Statement, Method, Discussion, Conclusion, Result, Related Work, Limitations, Further Work, Problems, Implementation, Example, Experiment, Evaluation, Data* and *Solution*. Pattern matching of a range of expressions in the headlines is applied. If no pattern matches, the value *NonPrototypical* is assigned.

5.2.1.4. Sentence Length

At first glance, the criterion of sentence length seems to be a trivial criterion which is not related to relevance or to argumentative zones. For trivial features, we expect a distribution which is near–identical to the global distribution of categories in the corpus, and therefore no help for a statistical classifier.

Kupiec et al. report better results when including the Sentence Length feature, but this point seems to be pertinent to their data coding: captions, titles and headings are not encoded as such and the sentence length feature can filter them out. In our corpus, this information is already directly encoded: sentence length thus cannot fulfill the filtering function.

But there are some other reasons why sentence length might *not* be a trivial feature after all. Sentence length is one indicator of sentence complexity which has been used in extraction experiments before. Earl (1970) argues that short sentences in her material are more likely to contain trivial material. Robin and McKeown (1996) state that complex sentences (conveying a maximal number of facts) are advantageous as a summary. There are, of course, other criteria for complexity apart from sentence length. Some measurements try to determine how contentful the sentence is by calculating the proportion of content words per length, or by measurements of the syntactic complexity of the sentence.

Sentence length might be a useful feature for Argumentative Zoning due to the high number of OWN sentences in our corpus, which describe details of the solution. They contain less meta-discourse than other sentences, and they tend to be less complex and thus shorter.

5.2.1.5. Syntactic Correlates of the Verb

In text extraction, there have been some efforts to use purely syntactic criteria for the indication of overall relevance, but most of these proved unsuccessful. Baxendale (1958) used the objects of prepositions as sole representation for the document. Earl (1970) describes an unsuccessful experiment to correlate global importance to the parts-of-speech (POS) shape of sentences. However, there were too many different POS shapes, and she concludes that:

> it seems fair to say that indexible and non-indexible sentences cannot be distinguished by structure alone. (Earl, 1970, p. 321)

Also interesting are experiments differentiating different linguistic factors per rhetorical sections. These experiments concentrate on the standard four-part fixed structure (*Introduction, Methods, Results, Discussion*), which is, as we have argued before, related to argumentative zones, albeit not in a trivial way (cf. section 3.1).

Verbal syntactic features can be indicators of rhetorical section structure, as studies like Biber and Finegan (1994) and Milas-Bracovic (1987) show. West (1980), for example, manually determined and counted the occurrence of *that*-nominals (e.g. *"the fact that…"*) in different rhetorical sections. That-nominals often indicate knowledge-stating sentences. West found that the density of that-nominals differed significantly between rhetorical sections: there were statistically more that-nominals in the *Introduction* and *Discussion* sections than in the *Results* section. The *Methods* section has fewer that-nominals than any other section.

Myers's (1992) work is particularly relevant to Argumentative Zoning. He describes properties of sentences stating authors' knowledge claims (our AIM sentences). Apart from two non-linguistic features (cue phrases and location), he lists the following linguistic features of the main verb in such sentences:

- Verb: *"to present", "to report"* or similar
- Tense: Present Perfect
- Person: First

We consider only verbal syntactic features here: voice, tense and the existence of a modal auxiliary.

Voice (`Syn-1`): Riley's (1991) work shows that there is a correlation between rhetorical roles and the use of the passive tense. The explanation for this is that voice is connected to *authors' perspective*. Prescriptive accounts of academic writing advise writers to avoid the mention of the own person, in order to avoid the impression that they are unduly interested in the success of their own research. This results in a high proportion of passive sentences, and often makes texts less readable and more difficult to understand. If a text is written in this style, it is sometimes difficult to tell who performed a certain research action. Many authors in our collection use the active voice instead to describe their own work, but nevertheless, there are also articles which use the passive voice frequently.

Tense (`Syn-2`): It has been hypothesized that authors use different tenses for different rhetorical segments (Biber and Finegan, 1994; Milas-Bracovic, 1987) or for certain argumentative tasks. Aspect and tense have been shown to correlate with discourse structures (Salager-Meyer, 1992; Hwang and Schubert, 1992; Malcolm, 1987). The connection between aspectual information (which is predominantly expressed by tense in English) and argumentation is that aspect signals the state of an activity ("*has the problem been solved or is it unsolved yet?*"). For example, the present perfect, being used for unfinished states, is often associated with pending problems, whereas the use of past tense, particularly in combination with statements of solution-hood, signal an accomplishment, i.e. the fact that an end state has been reached.

Another reason why tense should be an interesting feature for Argumentative Zoning is that many formal guidelines for publication, e.g. in certain journals, require authors to use past tense for descriptions of previous work, including own previous work, and present tense for current work. This distinction, as it is connected to the attribution of ownership, is particularly important for Argumentative Zoning. On the other hand, many of the authors in our collection are non-native speakers and might use tense in an idiosyncratic way.

Modality (`Syn-3`): The use of modal auxiliaries is one of the correlates for a phenomenon called *hedging* (cf. Hyland's (1998) hedging category in figure 3.13, p. 100). Hedging occurs when authors distance themselves from a scientific statement (Salager-Meyer, 1994). Other correlates of hedging are adverbials like *likely, possibly, maybe* which formed part of Edmundson's negative cue phrases. Hedging has been proposed as a signal for rhetorical sections, as it is associated with speculative statements in *Discussion* sections. Wiebe (1994) also uses the occurrence of a modal other than "*will*" for her subjective/objective distinction.

5.2.1.6. Citation Features

Type of Citation (`Cit-1`): Citations are a good indication that the topic of the sentence is somebody else's work; our human annotators use this factor to distinguish between OTHER and BACKGROUND categories. Thus, the existence or non-existence of formal citations should prove useful for Argumentative Zoning. We also believe that mentions of other authors' names in the text, even if these do not occur in a formal citation context, have a status similar to full citations. Consider sentence **8** of our example article:

> *In Hindle's proposal, words are similar if we have strong statistical evidence that they tend to participate in the same events.* (S-8, 9408011)

The full citation was used in sentence **5**; similarly to the use of pronominal reference, use of the author's name avoids repetitiveness. We think that in this sentence should be logically treated as if it had read "*In Hindle's (1993) proposal*", i.e. as if a formal citation had been present.

Self Citations (`Cit-2`): If some own previous work is mentioned in a paper, it is very likely that the authors mention it because they base their own work on it (BASIS). Therefore, the fact that previous work is the author's own should be recognized.

Citation Location (`Cit-3`): Citations are authorial if they form a syntactically integral part of the sentence, or parenthetical if they do not (Swales, 1990). We believe that the attribution of intellectual ownership is more often expressed by authorial citations, and that parenthetical citations are often there for other reasons (“piety, policy, politeness” cf. Ziman (1969)). If this is true, the syntactic type of a citation might prove useful for Argumentative Zoning.

As authorial citations form the subject of the sentence, they typically occur in the beginning, whereas most of the parenthetical uses of citations occur in the end of the sentence. Citation location (`Cit-3`) captures exactly this aspect.

5.2.2. Meta-Discourse Features

Meta-discourse represents one of the most reliable indicators of rhetorical status and is potentially very useful for Argumentative Zoning. Other computational approaches (Marcu, 1997a; Litman, 1996) also exploit meta-discourse, but meta-discourse of a different kind: short cue phrases belonging to a closed-class vocabulary (e.g. adverbials, sentence connectives or general relevance markers like “*in sum*”). As a result, the linguistic realization of such meta-discourse phrases tends to be invariant between disciplines and authors.

But when we looked at realizations of scientific meta-discourse in section 3.2.5, we found that apart from formulaic, fixed meta-discourse (*“to my knowledge”, “in this paper”*), there is another kind of meta-discourse which shows a wide range of syntactic variation—recall the different ways of expressing intellectual ancestry exemplified in figure 3.14 (p. 102). It is difficult to see how this type of meta-discourse could be captured with a fixed list; a more flexible way of analyzing it is needed.

We suggest that one way out of the dilemma of linguistic variation is to discover prototypical *agents* and *actions* individually in a wider range of syntactic contexts, e.g. in passive and active constructions (Teufel and Moens, In Prep.). Looking at the examples for the argumentative moves in figures 3.7, 3.9, 3.10, 3.12 and 3.15, one cannot help noticing that scientific argumentative text abounds in prototypical agents and actions, which recur in different syntactic disguises. We argue that it should be enough for Argumentative Zoning to recognize these prototypical actions and agents, while reading over all agents and actions that are not understood (and which are likely to refer to the science in the paper). As the patterns themselves are rather prototypical (*“our approach”*), pattern matching and syntactic heuristics should be able to find a large part of these agents and actions.

This would provide a simple profile of the agent/action structure of the document: the information of “who-does-what”. We assume that the agent/action structure is an integral part of the kind of document structure that we are looking for, and should help us perform Argumentative Zoning. We also believe that the agent/action structure provides a deeper, more semantic–oriented kind of text representation than the text strings themselves. Such intermediate representations have been called for by Spärck Jones (1999) as a prerequisite for better text summarization strategies.

One last caveat: the phrases we call meta-discourse *can* have a meta-discourse interpretation—but they do not always have this interpretation. Litman (1996) uses machine learning to address the problem that the phrase *"so"* can function as meta-discourse or as propositional contents. There are some ambiguity problems associated with our approach, which we discuss in section 6.2.

5.2.2.1. Formulaic Expressions (`Formu`)

The Formulaic Expressions Feature is designed to determine and classify explicit meta-discourse statements of a fixed kind.

Indicator or cue phrases have a long history as features for text extraction, i.e. for determining global sentence importance. In Edmundson's (1969) approach, sentences containing positive cue phrases like superlatives or explicit markers of importance or confidence ("*important*", *"definitely"*) were considered fit for extraction, whereas other sentences containing *stigma* words like *"hardly", "unclear", "perhaps", "for example"* (belittling expressions, expressions of insignificant detail or speculation/hedging) were discouraged from extraction. Edmundson's list was statistically acquired and manually corrected. A similar but much more extensive list containing 777 terms (called the Word Control List or WCL) was used in ADAM, the first commercially used automatic abstracting system (Pollock and Zamora, 1975).

More recent work on longer indicator phrases has been done by Paice and colleagues (Paice, 1981; Paice and Jones, 1993; Johnson et al., 1993), whereby sentences containing explicit rhetorical markers like "*the purpose of this research is*" or "*our investigation has shown that*" are considered fit for extraction. Paice (1981) describes the first implementation of a pattern-matching extraction mechanism relying on indicator phrases. Paice and Jones (1993) make the method more flexible by supplying a finite state grammar for indicator phrases specific to the agriculture domain; however, Oakes and Paice (1999) state that importance cues are often not reliable.

All these approaches use indicator phrases which indicate *global sentence relevance*—again, using indicator phrases for the determination of argumentative status is different. For example, the phrase "*in this paper, we have ...*" is a very good overall relevance indicator: it is quite likely that a sentence or paragraph starting with

Formu: Formulaic Expression Types			
Type	Example	Type	Example
GAP_INTRODUCTION	*to our knowledge*	PREVIOUS_CONTEXT	*elsewhere, we have*
OUR_AIM	*main contribution of this paper*	FUTURE	*avenue for improvement*
TEXTSTRUCTURE	*then we describe*	AFFECT	*hopefully*
DEIXIS	*in this paper*	PROBLEM	*drawback*
CONTINUATION	*following the argument in*	SOLUTION	*insight*
SIMILARITY	*similar to*	IN_ORDER_TO	*in order to*
COMPARISON	*when compared to our*	POSITIVE_ADJECTIVE	*appealing*
CONTRAST	*however*	NEGATIVE_ADJECTIVE	*unsatisfactory*
DETAIL	*this paper has also*	THEM_FORMULAIC	*along the lines of*
METHOD	*a novel method for X-ing*	GENERAL_FORMULAIC	*in traditional approaches*

Figure 5.6: Formulaic Expression Types (Feature Formu)

it will carry important discourse-level information. However, without knowing the following verb, we cannot be sure about the argumentative status of the sentence. It could continue with "... *used machine learning techniques for* ...", in which case the sentence is likely to be a description of solution/methodology; with a different verb, it might also be a conclusion ("... *argued that* ...") or a problem statement ("... *attacked the hard problem of* ...").

Our argumentative model in section 3.2 describes typical statements about the problem-solving processes in research. Our method for finding meta-discourse is to use pattern-matching on expressions that are expected by the model of argumentation introduced in section 3.2. We particularly concentrate on those meta-discourse expressions which have become formulaic expressions of scientific writing (cf. Hyland 1998; Swales 1990).

Our formulaic expressions are bundled into 20 major semantic groups. Figure 5.6 gives examples for the types of formulaic expressions used in feature Formu. For example, a marker like *"our goal in this paper"* is expected to co-occur frequently with the AIM category, whereas *"in the following section"* is a good marker for TEXTUAL. On the other hand, if we find a negative polarity item in the sentence e.g. "*however*"; *"no method has..."*; *"none of the approaches..."*, this raises the probability that we are dealing with a sentence which indicates a flaw of some other work (CON-

TRAST). Another good indication of a gap in knowledge is the phrase *"to our knowledge"*. The full list of 396 formulaic patterns is given in appendix D.1.

5.2.2.2. Agentivity Features (`Ag-1` and `Ag-2`)

The recognition of prototypical agents and actions serves to identify scientific meta-discourse which is less fixed than the phrases covered by the `Formu` feature. For writing styles that do not use much meta-discourse it might be particularly advantageous to determine agents and actions, because they might provide the only superficially marked correlates of argumentative status. For data collections with large variations in meta-discourse like ours, it makes sense to *classify* the agents and actions. Then it does not matter which particular term the authors use (e.g. *"we"*, *"I"* or *"one of us"*)—these expressions are represented as the same entity (US_AGENT), and automatic processing can generalize over the same concept.

Possibly the closest related work with respect to agents and actions is that of Barzilay et al. (1999), which uses overlap of actions and agents to detect the similarity of events in newspaper paragraphs. However, whereas in our text type *prototypical* agents are particularly relevant, in their text type (news stories), any potential agent needs to be matched.

In our approach, agents and actions are expressed separately and modularly; their syntactic context is recognized (passive vs. active), and negation is automatically taken into account. Such an approach is more robust and less error-prone than standard pattern matching methods which are string-based, as individual subject–verb combinations might easily be forgotten from such lists.

Using syntactic constraints in Agentivity features (i.e. agents and actions) also increases the precision of pattern matching. As an example, GAP_AGENT pattern are designed to find statements expressing the lack of a solution (*"no papers/articles/studies describe a solution to the problem…"*). But when GAP_AGENT patterns (e.g. *"no articles"*) are applied without syntactic restrictions (i.e. anywhere in the text), the error rate is high: 5 out of the 13 GAP_AGENT occurrences in our corpus were erroneous. The problem is polysemy: *"article"* can mean article-in-a-journal (the interpretation intended here), or it can also mean the grammatical article (*"a"* or *"the"*). If we, however, search for GAP_AGENT patterns only in subject positions (as determined by our heuristics), we reduce the error due to polysemy completely, and we get 9 out of 9 occurrences with the correct meaning.

For the practical implementation, we made the decision to give grammatical subjects (or by-objects in passive sentences) a special status by encoding them in feature `Ag-1`; we disregard grammatical patients (typically direct objects) even though in many cases the information contained in objects is potentially relevant too (*"we solve the problem of..."*). However, we feel that the robust recognition of subjects (agents) and semantic verbs (agents), as in our approach, is a workable middle ground between shallow and deep text representation.

Agents (`Ag-1`**):** Agent-hood should be a good indicator of Argumentative Zoning, as it is related to attribution of authorship, which is a defining factor in basically all of our categories. The main agent groups are US_AGENT, GENERAL_AGENT and THEM_AGENT.

Authors often have to refer to themselves; we call this agent class US_AGENT. The terms *"I", "we"* and *"the first author"* all refer to this class. Personal pronouns in 1st person ("*I*" and "*we*") are an important help. The Roman number 1, can, however, be mistaken for the pronoun "*I*", as in the following erroneous example:

> <AGENT TYPE="US_AGENT"> **I** </AGENT> **is an interpretation iff** <AGENT TYPE="US_AGENT"> **I** </AGENT> **is a triple** <EQN/>
> (S-21, 9408003).

As we do not check for subject-verb agreement, such errors cannot be avoided in our processing, but they do occur only rarely.

There are also cases where the explicit marking of agenthood might be deceptive. A sentence starting with "*we*" might occasionally have a different function from describing own work. It might be used to clarify notation, to draw preliminary conclusions, to direct the attention of the reader to some non-obvious fact or to explain the presentational form in which an idea (possibly attributed to somebody else) will be presented in the article.

For example, authors might state in one sentence that researcher X has introduced a particular algorithm. The next sentence might state that "*We will demonstrate how the algorithm works by way of example*"—followed by a long (unmarked) description of the algorithm. It is clear to humans that these sentences are attributed to X, and not to the authors. A simple algorithm which assumes that non-marked sentences always carry the status that the last marked sentence displayed will, however, lead to the wrong guess that the long segment is attributed to the authors.

Distinguishing previous own work from the current approach is a difficult case. After such previous own work has been introduced with a self citation, most authors use a 1st person pronoun to refer to it, but some authors use a 3rd person pronoun (particularly if the cited paper is co-authored). However, we found no 3rd *singular* pronominal reference to own previous work in our corpus. The use of 3rd person pronomina might have to do with the instructions for double-blind reviewing of papers: The instructions specifically state that citations of own previous work should not reveal the identity of the author, and many authors obviously did not not change the pronomina after the paper was accepted.

There is a real problem if the description of own previous work is directly followed by a description of the current work in the paper, and if the authors do not use an explicit formulaic signal ("*in this paper*"). In this case, it is almost impossible to guess where in the text "*us*" stops to mean "*us, previously*" and begins to mean "*us, now*".

Noun phrases with a possessive 1st person determiner ("*our*" or "*my*") also indicate own work, if the head of that noun phrase is a prototypical solution (e.g. *"theory, approach, method, algorithm"*), as the authors' approach or solution is often equated with the players "US". The solution type list is also used for the METHOD pattern above in `Formu`. Our list of solution nouns is given in appendix D.4.

When trying to find mentions of "THEM_AGENT" in text, the following patterns lend themselves well:

- Authorial citations are the best indication of a THEM_AGENT.
- The names of other researchers is an equally good indication of a THEM-AGENT. In our implementation, author names are recognized and are annotated before processing.
- 3rd person possessive pronoun plus solution nouns ("*their system*").
- Personal 3rd pronouns *can* refer to THEM_AGENTs, particularly after formal references (and if the grammatical number is right). However, 3rd person personal pronouns might just as well refer to other things: Singular pronouns often refer to fictional characters in the example sentences. The plural pronoun "*they*" can refer to any plural object in the research world, e.g. rules, formulae or trees.
- A demonstrative pronoun plus a solution noun (*"this approach"*) is ambiguous between a reference to US_AGENT and to THEM_AGENT.

When trying to find mentions of "THEM-GENERAL" in text, the patterns we are looking for are quite formulaic.

- Some expressions follow the pattern "*general people in the field*". We use a list of professions, e.g. "*workers, linguists, computer scientists, researchers...*" and allow for syntactic variations, e.g. modification with typical adjectives.
- Other expressions follow the pattern "*previous papers*". We use a list of entities like "*article, paper, work, research*" and allow for syntactic variations. All these groups of nouns can be found in appendix D.4.
- Yet other expressions are variations of the pattern "*traditional solutions in the field*". We use the aforementioned list of solution types.

Figure 5.7 lists the agent types we distinguish. Rather than just the agent types US_AGENT, THEM_AGENT and GENERAL_AGENT and a fourth type US_PREVIOUS_AGENT, there are altogether 13 types. Some of these are non-personal (pseudo) agents like aims, problems, solutions, absence of solution, or textual segments: OUR_AIM_AGENT; PROBLEM_AGENT; SOLUTION_AGENT; GAP_AGENT; TEXTSTRUCTURE_AGENT (*"this section"*). In other agent types the syntactic form does not allow to determine the referent unambiguously, e.g. because of pronominal

Ag-1: Agent Types	
Type	Example
US_AGENT	*we*
REF_US_AGENT	*this paper*
OUR_AIM_AGENT	*the point of this study*
AIM_REF_AGENT	*its goal*
US_PREVIOUS_AGENT	*the approach given in* <REF SELF=YES/>
REF_AGENT	*the paper*
THEM_PRONOUN_AGENT	*they*
THEM_AGENT	*his approach*
GAP_AGENT	*none of these papers*
GENERAL_AGENT	*traditional methods*
PROBLEM_AGENT	*these drawbacks*
SOLUTION_AGENT	*a way out of this dilemma*
TEXTSTRUCTURE_AGENT	*the concluding chapter*

Figure 5.7: Types of Agents (Feature Ag-1)

or deictic anaphora (*"this approach"*). Such forms are clustered together into ambiguity classes with a lower confidence level: REF_US_AGENT, THEM_PRONOUN_AGENT, AIM_REF_AGENT and REF_AGENT. The 168 agent patterns we use are given in appendix D.2 (p. 339).

It is possible that the agent patterns appear in a position other than subject position, in which case they still carry some information, even if they are not the agents. In this case, they are reported under the `Formu` feature; the 13 `Ag-1` classes are thus added as values to the 20 `Formu` types, resulting in a total of 33 values for the feature `Formu`.

Actions (`Ag-2`): This section discusses a classification of verbs into semantic classes which assist Argumentative Zoning. Verbs are not frequently used in NLP experiments, in contrast to nouns. Klavans and Kan (1998) are an exception in that they use verbal classes for document classification according to text type and event. They use Levin's (1993) alternation classes and found that occurrence of communication verbs and agreement verbs correlated with text type and/or event (e.g. opinion pieces vs. documents about legal cases or mergers). In contrast to ours their work looks at large text units (documents) whereas we are interested in using verb information per sentence.

Negation is a phenomenon which should be recognized—there is an essential difference between the action of "*does not solve*" and "*solves*". Not understanding this difference would deliver the opposite interpretation to the one intended and thus undermine the core of our shallow selective text-understanding task. We heuristically determine if a verb is negated or not.

We use a manually constructed verb lexicon for verb classification, cf. figure 5.8. The semantics of these verbs mainly comes from the argumentative moves defined in section 3.2, which are concerned with similarity, contrast, competition, presentation, argumentation and textual structure. We will describe them in the following:

PRESENTATION_ACTIONs include verbs like *present, report, state*, often referred to as communication verbs. Myers (1992) performs a pragmatic analysis of such verbs in combination with knowledge claims; Thomas and Hawes (1994) analyze such verbs in medical texts, and Thompson and Yiyun (1991) look at presenting verbs in the context of citations and positive/negative evaluation.

Explicit signalling of the research process ahead is another frequent phenomenon. Research goals can be introduced by stating an interest in a certain research question (INTEREST_ACTION; *"aim to", "attempt to"*) or by stating some involvement or affect towards the solving of a problem (AFFECT_ACTION; *"seek", "want"* and *"wish"*). Direct argumentation verbs (ARGUMENTATION_ACTION) include *"argue", "disagree"* and *"object to"*.

In statements about problem-solving processes (cf. section 3.2.4), verbs of problem introduction abound (PROBLEM_ACTION). These are the ones which state that a situation is problematic. Examples for verbs in this class are *"fail", "degrade", "overestimate",* and *"waste"*. If there is a lack or need of something, this often has the same semantics (NEED_ACTION; verbs like *"lack", "need", "be void of"*). Problem-solving actions (SOLUTION_ACTION) indicate that a solution has been found (*"solve", "circumvent", "mitigate"*). Contrast between approaches might be expressed overtly with CONTRAST_ACTION verbs like *"clash", "contrast with"*, and *"distinguish"*. BETTER_SOLUTION_ACTIONS state that one solution solves the problem better than another. Examples include *"outperform"* and *"increase"*). Comparison actions (COMPARISON_ACTION) draw a direct comparison between own and rival approaches (*"compare with", "test against"*). Display-of-awareness verbs (AWARENESS_ACTION) like *"know"* can be used to show that there is a gap in the literature, or that the own task is done for the first time, as in the phrase *"we know of no approach which…"*.

There is a range of ways of stating that aspects of a solution are borrowed from another one. CONTINUATION_ACTIONS include *"base on", "borrow", "take as our starting point"*. Another way of stating research continuity is to state the simple use of another solution (USE_ACTION; *"employ", "use"*); this can be combined with a statement of which aspect of the other solution was changed (CHANGE_ACTION; *"transform", "change"*). In some cases, similarity between solutions (SIMILARITY_ACTION) is stated as a signal for intellectual ancestry (*"resemble", "be similar"*).

There are generic, prototypical RESEARCH_ACTIONS which can be predicted from the discipline (e.g. *"analyze", "conduct", "define"* and *"observe"*). Many other such actions are document specific, describing the creative inventive step of the article. They can therefore not be predicted. We also look for TEXTSTRUCTURING_ACTIONS such as *"outline"* and *"structure"*.

The action lexicon contains a total of 365 verbs; it is reproduced in appendix D.3 (p. 343). This lexicon also contains phrasal verbs and longer idiomatic expressions (e.g., *"have to"* is a NEED_ACTION; *"be inspired by"* is a CONTINUE_ACTION).

Ag-2: Action Types			
Type	Example	Type	Example
AFFECT	*we hope to improve our results*	NEED	*this approach, however, lacks...*
ARGUMENTATION	*we argue against a model of*	PRESENTATION	*we present here a method for...*
AWARENESS	*we are not aware of attempts*	PROBLEM	*this approach fails...*
BETTER_SOLUTION	*our system outperforms ...*	RESEARCH	*we collected our data from...*
CHANGE	*we extend <CITE/>'s algorithm*	SIMILAR	*our approach resembles that of*
COMPARISON	*we tested our system against...*	SOLUTION	*we solve this problem by...*
CONTINUATION	*we follow Sag (1976) ...*	TEXTSTRUCTURE	*the paper is organized...*
CONTRAST	*our approach differs from ...*	USE	*we employ Suzuki's method...*
FUTURE_INTEREST	*we intend to improve ...*	COPULA	*our goal is to...*
INTEREST	*we are concerned with ...*	POSSESSION	*we have three goals...*

Figure 5.8: Types of Actions (Feature Ag-2)

5.3. A Prototype System

We have implemented a statistical and a symbolic Argumentative Zoning prototype system. Our corpus is encoded in XML (eXtensible Markup Language). XML, which provides a universally recognized platform for data representation, also allows the definition of customized semantic labels. This helps in the encoding of the document's semantics, rather than just layout information.

Processing is based on a Unix pipeline. Different phases of the pipeline add different information (in the form of XML elements and attributes) to an intermediate XML representation of the document.

The corpus collection and conversion work was initially conducted in summer 1996 by myself and Byron Georgantopoulos, as a joint effort to provide data for different projects with the summarization of academic papers. The final conversion pipeline uses a different implementation, based on the TTT tools available from the HCRC Language Technology Group (Grover et al., 1999). A version of the corpus collected during the current work is now available from Tipster SUMMAC (1999).

5.3.1. Corpus Encoding

The first step in the endeavour to collect a corpus is the design of a corpus encoding format. On the one hand, one wants to encode as much of the original information as possible. It is desirable to standardize the encoding such that it expresses the document semantics, and abstract away from the physical and typesetting information the data comes mixed with. Our XML encoding provides rich information about structural information, e.g. sentences, paragraphs and division structure. The author-written summary is marked as such. Additional mark-up includes titles, headlines, sentences, formal citations, author names and the reference list at the end.

Another criterion is data consistency. LaTeX, the source encoding of our data, is unfortunately a very powerful language, offering a wide range of syntactic constructs. Therefore, similar document semantics might be expressed syntactically differently in different papers (in the worst case even in the same paper), but our encoding should treat them alike.

The two goals of information-richness and data consistency often work against each other. For example, citation handling can be automated in LaTeX with the command `\cite`, but authors could decide to just type the author name and year. Similarly, cross references can be expressed with the command `\cref`; however, some authors prefer to directly state the actual numerical cross reference. Ideally, our representation should mark up both facts: the fact that the string "*2.2*" refers to a cross reference (type information), and that its identity is "*2.2*" (string information). However, if authors used `\cref`, we do not have the identity of the string (as it is only determined at run-time of the LaTeX system), whereas the textual variant does not give us the information that the string's type is a cross reference. We decided to use the structural information in preference to the string information—in general, we preferred consistency above informativeness in conflict cases. This means that in our encoding type/structural information is captured consistently, however sometimes at the price of a small information loss.

There are some design decisions which were influenced by the fact that corpus collection took place in collaboration with a project that was less interested in structural features than the current thesis is. The loss of captions is an example of a wrong but non-reversible design decision. It was decided in an early processing stage to remove captions of images and tables. Part of the reason for doing so was data consistency, as captions cannot always be determined automatically. We realized only later that captions often contain information particularly useful for summarization.

It was also decided to remove footnotes, a decision which we do not regret. As textual material contained in footnotes is marked by the author as less central to the overall flow of the argumentation, a summarization system might decide to ignore it. However, for a full representation of a paper, which is not attempted here, footnote text should be kept. Footnote information might be important if one tries to assess relative importance of citations, as some marginal references appear only in footnotes.

Appendix B.1 shows the example paper in XML format after preprocessing, before feature determination. We will now describe in detail how the document semantics of the papers are encoded in XML. Appendix A.1 gives the DTD (*Document Type Definition*) for our corpus. A DTD is a BNF-style description of the hierarchical and logical structure of an XML file. As DTD syntax is cryptic and might be unknown to the reader, the following list explains the components in English.

- *Title, authors and bibliographic information* is marked by elements <TITLE>, <AUTHOR>, <AUTHORS>, <FILENO>, <APPEARED>
- A *unique citation form* is assigned to the document and marked as <REFLABEL>. The citation form is a mnemonic label consisting of name and date, and of an optional letter to distinguish references which are ambiguous within the corpus, if needed. The provision of unique citation forms is important for disambiguation of citations (e.g. for clustering of documents by bibliographic chaining).
- *Divisions:* The hierarchical embedding of text segments is encoded by the <DIV> element, which is recursive. The DEPTH attribute indicates the depth of embedding of a division. Each division must start with a <HEADER> element.
- *Headlines* are marked as <HEADER> elements, containing (tokenized and POS-tagged) text.
- *Appendices:* If appendices occur at some other place in the paper, they are physically moved to the point directly before the reference list. They do not receive preferential treatment; instead, they are treated like all other divisions. The fact that they are appendices can only be read off the headline.
- *Paragraphs:* Paragraphs are marked as element <P>.

- *Sentences:* Sentences are separated and marked as <S> elements. This is important, as sentences are the base level selection and analysis unit.
- *Abstract:* The abstract is marked as <ABSTRACT>, and sentences of the abstract are marked as elements <A-S>.
- *Correspondences* between abstract and document sentences are marked by a double link: attribute DOCUMENTC in abstract sentences, and attribute ABSTRACTC in document sentences. This correspondence is determined by a similarity finding algorithm and manual checking (cf. section 4.1.2.2).
- *Images*: Images are removed and the place is marked by an empty <IMAGE/> element. In cases where the LaTeX `verbatim` environment was used, it was manually decided whether or not such material counts as an image or as text.
- *Tables:* Tables are removed (often automatically, sometimes manually), and their position is marked by an empty <IMAGE/> element.
- *Bullet point lists*: Bullet items are manually marked up as such by as an optional attribute of sentences (TYPE=ITEM). Paragraphs as well as sentences can be bullet items.
- *Cross references:* Cross references are automatically or manually marked as empty elements <CREF/>. Manual effort was needed to find corresponding numbers (*"figure 1"*) and replace them by <CREF/>. For consistency reasons, we erased the numbers themselves, as they were not in all cases available.
- (Linguistic) *example sentences* are manually marked up as <EXAMPLE>.
- *Equations:* any kind of mathematical formula that could not be expressed in ASCII was manually (sometimes automatically) replaced by empty element <EQN/>. There might be cases of inconsistencies with formulas like P(A,B) which might be expressed as ASCII or as as <EQN/>, depending on whether the author used the LaTeX math mode or not.
- *Bibliography list:* During bibliographic processing, the bibliography list at the end is marked as <REFERENCE>. It consists of single <REFERENCE> items, each referring to a formal reference. Within these reference items, names of authors are marked as <SURNAME> elements, and years as <YEAR>.

- Formal *citations:* During preprocessing, formal citations are marked automatically as <REF> wherever the latex command `\cite` was used; otherwise, bibliographic processing automatically marks them. Self references are automatically recognized by comparing the names of the author(s) of the paper with all author names associated with the reference. They are marked using the attribute `SELF`.

- *Names of other authors:* Author names occurring in running text without a data are marked up as <REFAUTHOR> during the bibliographic processing step.

- *Formulaic expressions:* if formulaic expressions are recognized during feature determination, they are marked as <FORMULAIC>, with an attribute specifying the formulaic expression type.

- *Agents:* if prototypical agents are recognized during feature determination, they are marked as <AGENT>, with an attribute specifying the agent type.

- *Actions:* if prototypical actions are recognized during feature determination, they are marked as <ACTION>, with an attribute specifying the action type.

5.3.2. Preprocessing

We chose all papers from CMP_LG which fulfilled the following criteria:

- *Date:* We collected all papers put on the archive between 04/94 and 05/96.

- *Format:* The LATEXsource had to be available (in addition to a PostScript version of the paper), and the paper had to pass our conversion pipeline automatically; about 20% did not pass or showed too many errors such that manually correction would have been too inefficient.

- *Abstract:* The papers had to have an abstract.

- *Type:* The papers had to be published in the proceedings of the main or student session, or of a workshop of one of the following conferences: *The Annual Meeting of the Association for Computational Linguistics* (ACL), *The Meeting of the European Chapter of the Association for Computational Linguistics* (EACL), the *Conference on Applied Natural Language Processing* (ANLP), and the *International Conference on Computational Linguistics* (COLING).

As a result of being published in conference or workshop proceedings, the length of the papers was restricted by the publishing rules of the corresponding proceedings. The PostScript versions of the papers are between 3 and 10 pages long; most papers are between 6 and 8 pages long.

The corpus consists of 333,634 word tokens (counting punctuation as a token), the average number of tokens per paper was 4170, ranging from 1301 to 7635 tokens. The total number of document sentences is 12471, average per paper is 156, ranging from 45 to 322. The total number of abstract sentences is 356, average per paper is 4.5, ranging between 2 and 13 sentences.

Our papers' original format was LaTeX source. The first processing steps are a text format conversion from LaTeX source to XML format: LaTeX source is converted into HTML with the program Latex2html (Drakos, 1994; Latex2Html, 1999); the resulting HTML format is then transformed into XML format with a range of `perl` scripts. The pipeline is fully implemented, but some manual correction effort is still needed as the pipeline works imperfectly. This is due to the difficulty of deducing semantic markup from layout information:

- LaTeX is a rich language, offering a wide range of syntactic constructs which are difficult to standardize.
- Latex2html has certain weaknesses, e.g. the inability to deal with LaTeX macros.
- Our XML encoding contains some information which no automatic processing can perform yet (e.g. the determination of (linguistic) example sentences in text).

As a result of the preprocessing/conversion step, text is in a format in which paragraphs are marked up, but words are not separated yet, and sentences are not marked either. The next step is a pipeline to provide linguistic mark-up, and to determine the values of the features, as described in the next section.

5.3.3. Feature Determination

We will now describe how features are automatically determined in running text. Figure 5.9 shows the single steps of processing; it also shows which feature values each processing step provides.

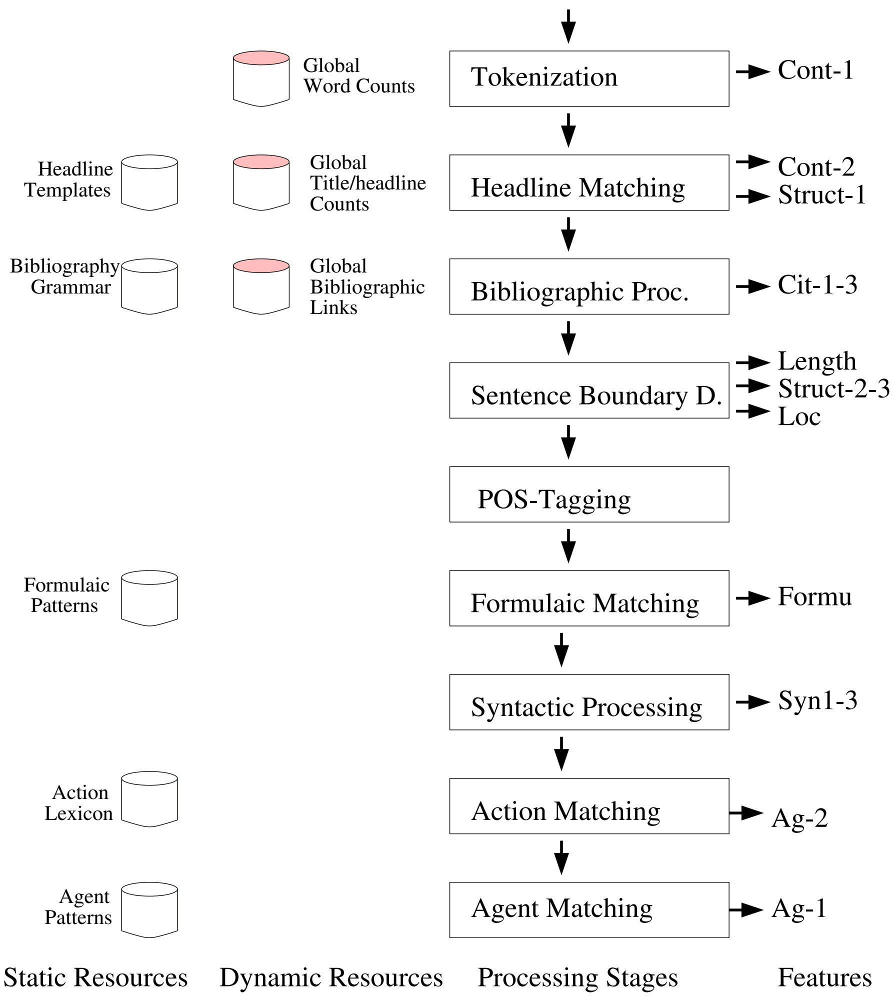

Figure 5.9: Feature Determination Steps

We will describe the practical algorithm for determining the value for each feature. We will also give contingency tables for each feature. Whenever 100% correctness of a feature cannot trivially be assumed, we have also performed an evaluation of the reliably of the heuristics used.

5.3.3.1. Tokenization

Tokenization is the first step in our feature determination pipeline. We used software distributed as the TTT (Text Tokenization) System by the HCRC Language Technology Group Grover et al. (1999). The tokenization grammar was written by Claire Grover; it performs separation of word tokens from the ASCII stream. Tokenization provides information needed for feature `Cont-1`.

`Cont-1`	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
0	129	193	658	537	1801	7517	172	11007
1	78	33	62	59	213	919	51	1415
Total	207	226	720	596	2014	8436	223	12422

Figure 5.10: Contingency Table for *tf/idf* Feature (`Cont-1`)

In order to calculate the *tf/idf* score $w_{i,j}$, we use the following formula:

$$w_{i,j} = f_{i,j} * log(\frac{N}{n_i})$$

$w_{i,j}$:	weight for a word k_i in document d_j
n_i:	number of documents containing word k_i
$f_{i,j}$:	frequency of word k_i in document d_j
N:	number of documents in collection

The n top-scoring words according to the *tf/idf* method are chosen as content words; sentence scores are then computed as a weighted count of the content words in a sentence, meaned by sentence length. The m top-rated sentences obtain score 1, all others 0. We received best results with $n = 10$ and $m = 40$. The contingency table is given in figure 5.10.

5.3.3.2. Headline Matching

Headlines are used for two features in our implementation, `Struct-3` and `Cont-2` (cf. figures 5.11 and 5.12 for contingency tables).

For the feature `Struct-3`, we pattern match the headline against 89 patterns which correspond to 16 prototypical headlines. If there is a hierarchical nesting of divisions, the headlines of the deeper embedded sections are considered first. If no pattern matches, the value `Non-Prototypical` is assigned. We can see that more than 45% of all sentences (5576/12422) are not covered by prototypical section headings, i.e. they cannot be easily associated with a rhetorical section. This is in agreement with our argumentation in section 3.1.

`Cont-2` is the title method. In our implementation, title scores are determined as the mean frequency of n (or less) title word occurrences (excluding stop-list words). If the title contains more than n non-stoplist words, the n top-scoring words according to the *tf/idf* method are chosen. Again, the m top-scoring sentences receive the value 1, all other sentences 0. Best results in this case were received with n=10 and m=18. One

Struct-3	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
Introduction	102	48	382	185	434	368	89	1608
Implementation	1	18	5	24	262	791	9	1110
Example	1	10	16	27	112	459	6	631
Conclusion	62	14	4	39	27	454	3	603
Result		2		7	33	480	6	528
Evaluation	4	3	1	10	27	427	5	477
Solution	1	7	18	21	78	280	4	409
Experiment		11	4	9	19	306	1	350
Discussion	4	4	3	19	19	277	7	333
Method	1	7	4	26	40	163	6	247
Problems	3	7	14	9	20	95	1	149
Related Work	2	3	5	41	75	19	1	146
Data		1			6	102		109
Further Work					1	71		72
Problem Statement	1	1	5	1	2	42		52
Limitations		1	1	4	9	5	2	22
Non-Prototypical	25	89	258	174	850	4097	83	5576
Total	207	226	720	596	2014	8436	223	12422

Figure 5.11: Contingency Table for Headline Feature (`Struct-3`)

Cont-2	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
0	128	161	571	437	1546	6201	178	9222
1	79	65	149	159	468	2235	45	3200
Total	207	226	720	596	2014	8436	223	12422

Figure 5.12: Contingency Table for Title Feature (`Cont-2`)

variant of the method additionally takes words occurring in all headlines into account, but we received better results using only title words.

5.3.3.3. Bibliographic Processing

Bibliographic processing determines information important for features `Cit-1`, `Cit-2` and `Cit-3`. For the bibliographic processing we used a grammar written in the specific syntax of the program `fsgmatch`, which is provided with TTT. The grammar was originally written by Colin Mattheson; we changed it to suit our purposes. Bibliographic processing includes the following processing:

- The reference list at the end is parsed according to a grammar for bibliographic entries. This grammar anticipates typical citation styles. Author names and dates are marked up as such, and a <REFLABEL> element is constructed for each bibliographic entry, based on this information.

- The last names of all cited authors are put into a special lexicon, and the body of the text is searched in a second pass for these names.
- If the last names appear in a typical citation context (i.e. with a year, with or without brackets), they are wrapped as XML-elements <REF>. If they occur on their own, they are marked as <REFAUTHOR>. If the LaTeX command `\cite` was used, nothing needs to be done, as <REF> elements are already marked.
- Each reference is checked for overlap of one of the cited authors with the authors of the article (by comparison of all cited authors with the <AUTHOR> field). If such an overlap is determined, the reference is marked as a *self citation*. That means that the common abbreviation *"et al."* in citations in running text is resolved into all cited author names. This piece of information is only available from the reference list (even for human interpretation).

After all <REF> and <REFAUTHOR> in a sentence have been marked up, `Cit-1` reports the existence of either of these (if a sentence contains both <REF> and <REFAUTHOR>, the value `Citation` is chosen, cf. contingency table in figure 5.13). `Cit-2` reports whether or not a reference is a self reference, cf. contingency table in figure 5.14). In cases where a self citation and a non-self-citation appear in one sentence, the self citation is given preference. `Cit-3` gives the location of the reference(s) in order to distinguish authorial from parenthetical citations, cf. contingency table in figure 5.15. In cases of more than one reference in a sentence, "Citation-Beginning" is given preference over both "Citation-Middle" and "Citation-Ending", and "Citation-Ending" is given preference over "Citation-Middle".

`Cit-1`	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
Citation	17	163	79	96	482	290	5	1132
Author name	7	18	1	52	128	71	2	279
No Citation	183	45	640	448	1404	8075	216	11011
Total	207	226	720	596	2014	8436	223	12422

Figure 5.13: Contingency Table for Citation Feature (`Cit-1`)

`Cit-2`	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
Citation to Other Work	12	112	75	78	391	240	3	911
Citation to Own Previous Work	5	51	4	18	91	50	2	221
No Citation	190	63	641	500	1532	8146	218	11290
Total	207	226	720	596	2014	8436	223	12422

Figure 5.14: Contingency Table for Citation Type Feature (`Cit-2`)

`Cit-3`	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
Citation-Beginning		11	7	16	110	24		168
Citation-Middle	5	61	13	50	153	97		379
Citation-Ending	12	91	59	30	219	169	5	585
No Citation	190	63	641	500	1532	8146	218	11290
Total	207	226	720	596	2014	8436	223	12422

Figure 5.15: Contingency Table for Citation Location Feature (`Cit-3`)

5.3.3.4. Sentence Boundary Disambiguation

Determining sentence boundaries is important for each single feature, as sentences are our units of classification. However, some feature values can be determined directly after this step, namely the features `Length` (Sentence Length), `Struct-1` (Position in Section), `Struct-2` (Position in Paragraph), and `Loc` (Absolute Location).

We use the sentence boundary disambiguator provided with TTT (`ltstop`) and add some perl code to assign identifiers to sentences. We also had to write some code to mend some of the systematic mistakes the automatic method performed. We fixed such errors with symbolic rules. For example, in the following sentence the system failed to recognize a sentence break after a variable consisting of a single letter:

> <S> [...] *we make use of parameters ("dependency parameters") <EQN/> for the probability, given a node h and a relation r, that w is an r-dependent of* **h. Under** *the assumption that the dependents of a head are chosen independently from each other, the probability of deriving c is:< /S>*
>
> (S-190, 9408014)

Figures 5.16, 5.17, 5.18 and 5.19 give the contingency tables for features `Length`, `Struct-1`, `Struct-2` and `Loc`, respectively. For feature `Length`, the value 0 means that the sentence was shorter than a fixed threshold (here: 15 tokens including punctuation), 1 means that it was longer than the threshold.

`Length`	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
0	31	41	190	105	554	2507	102	3530
1	176	185	530	491	1460	5929	121	8892
Total	207	226	720	596	2014	8436	223	12422

Figure 5.16: Contingency Table for Sentence Length Feature (`Length`)

`Struct-1`	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
First_third	24	23	195	104	366	1174	22	1908
Second_third	36	48	190	169	736	2518	25	3722
Last_third	22	25	64	118	307	1600	27	2163
First_sentence	57	35	92	19	89	332	32	656
Last_sentence	15	14	7	25	51	487	40	639
Second_or_third_sentence	33	43	129	55	205	793	26	1284
Second-last_or_third-last_sentence	20	38	43	106	260	1532	51	2050
Total	207	226	720	596	2014	8436	223	12422

Figure 5.17: Contingency Table for Section Structure Feature (`Struct-1`)

`Struct-2`	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
Initial	117	92	267	135	601	2532	73	3817
Medial	56	87	306	289	971	3779	68	5556
Final	34	47	147	172	442	2125	82	3049
Total	207	226	720	596	2014	8436	223	12422

Figure 5.18: Contingency Table for Paragraph Feature (`Struct-2`)

For the feature `Struct-1`, the section is separated into three equally sized portions (measured in sentences). In those cases where a sentence is in a specific position within the section, the resulting values are "overwritten" over the tri-section values.

As far as feature `Struct-2` is concerned, if a paragraph contains only one sentence, that sentence receives the value `Initial`. If a paragraph contains only two sentences, the first sentence receives the value `Initial` and the second the value `Final`.

Values of the feature `Loc` are determined by dividing the sentence number of the document by 20, and assigning values according to the diagram in figure 5.5. Document areas corresponding to A, B, C, D, I, J are one twentieth of the document in length, E, G, H one tenth, and value F two fifth.

Loc	Aim	Bas	Bkg	Ctr	Oth	Own	Txt	Total
A	51	18	261	69	167	70	22	658
B	30	18	114	94	186	146	29	617
C	24	20	83	55	199	216	24	621
D	12	12	82	41	160	289	27	623
E	17	25	60	52	363	682	38	1237
F	7	81	104	178	680	3864	66	4980
G	2	11	12	21	121	1052	10	1229
H	6	19	1	30	62	1130	4	1252
I	23	11		31	43	514	2	624
J	35	11	3	25	33	473	1	581
Total	207	226	720	596	2014	8436	223	12422

Figure 5.19: Contingency Table for Absolute Location Feature (`Loc`)

5.3.3.5. POS-Tagging

Part of speech tagging provides vital information for complex pattern matching algorithms further on in the pipeline (Formulaic pattern matching, Agent Matching, Action Matching). It is performed using the program `ltpos`, distributed with TTT and written by Andrei Mikheev. It assigns one of the tags of the BROWN tagset (Francis and Kucera, 1982) to each token in text.

As later processing heuristics depend on the correct determination of finite verbs, we needed to determine the error rate of POS tagging. We manually checked the assignment of finite verbs, i.e. the tags VBP, VBZ and VBD on a random sample of 100 sentences containing finite verbs. We compared the automatic POS-tag with the POS-tag we thought should have been assigned. In the 100 sentences, there were 184 finite verbs, 174 of which the system recognized (recall of 95%). Most of the non-recognition errors were present verbs which the system erroneously tagged as singular or plural nouns. The system erroneously tagged an additional 14 tokens as finite verbs (precision of 93%). These words were mostly past participles in reduced relative clause constructions. We feel that this is a solid tagging performance, stable enough to base our further heuristic processing on it.

5.3.3.6. Formulaic Pattern Matching

We have determined a total of 396 formulaic patterns (cf. appendix D.1). As we use a finite-state replace mechanism, these patterns multiply out to many more actual strings. The lexical group of @TRADITIONAL_ADJECTIVES for example includes 37 adjectives like *classic* or *long-standing*, and this lexical group is contained in 29 patterns. There are 44 different lexical groups (cf. the concept lexicon appendix D.4). Some of the patterns use POS place-holders which are checked against the POS-tags of words in running text.

Additionally, the 168 agent patterns are also considered as formulaic patterns, wherever they do *not* occur as the subject of the sentence. The decision to include these into the `Formu` feature was explained in section 5.2.2.2.

Pattern matching procedures on such a large scale are slow. We reduce the number of comparisons necessary with a trigger mechanism: only to those sentences containing a trigger (a rare word which covers as many patterns as possible) are searched, and they are searched only for those patterns which do contain the trigger. Triggers are marked by the signal ↑ directly in the pattern.

Figure 5.20 gives the contingency table for `Formu`. It lists *first occurrence* of a formulaic pattern in the text. The restriction to one value per sentence is necessary for the Naive Bayes classifier.

5.3.3.7. Syntactic Processing

Syntactic processing determines the verbal features (`Syn-1, Syn-2, Syn-3`) and negation. It also determines the base form of the semantic verb, to be used for feature `Ag-2`. The first step of the algorithm is the determination of finite verbs in the sentence, information which is made available by the POS-Tagging. The next step is a finite state algorithm which checks left and right context of the finite verb for verbal forms of interest which might make up more complex tenses. Such forms are searched within the assumed clause boundaries, and additionally within a fixed window of 6 to the right of the finite verb. Negation is determined by a simple heuristic that searches for a list of 32 negation-items in the surrounding window of 5 items. The list of negation-items is given in appendix D.4 (p. 345).

The syntactic heuristics can contain errors, either due to errors in our algorithm or due to wrong POS-Tagging. We performed an evaluation on the aforementioned 100 sentences. Counting success and failure on the 174 finite verbs correctly determined by POS-Tagging, we found that the heuristics for negation and modality worked without any errors in our sample (100% accuracy), that there were 2 errors in the tense heuristics (99% accuracy) and 7 errors in the voice heuristics, 2 of which are due to POS-Tagging errors (where a past participle was not recognized in a passive sentence). The remaining 5 voice errors correspond to a 98% accuracy. Voice errors are particu-

`Formu`	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
GAP_INTRODUCTION				1	1	6		8
OUR_AIM	6					2		8
DEIXIS	1	1		2	3	45	3	55
SIMILARITY	2	3	1	1	7	4		18
COMPARISON		1		9	6	6		22
CONTRAST			11	41	17	100		169
DETAIL	1	1			1	36		39
METHOD	28	17	16	14	57	117	10	259
PREVIOUS_CONTEXT		1			2			3
FUTURE					1	20		21
AFFECT						6		6
PROBLEM			10	3	12	62		87
SOLUTION		1	7	4	29	81	3	125
IN_ORDER_TO	2	1	3	1	10	51		68
POSITIVE_ADJECTIVE	27	23	86	88	185	936	16	1361
NEGATIVE_ADJECTIVE	11	9	65	133	143	680	2	1043
THEM_FORMULAIC					4	1		5
AIM_REF_AGENT	13	2	20	7	26	121	2	191
TEXTSTRUCTURE_AGENT	2	3			5	21	83	114
GAP_AGENT				1		3		4
REF_AGENT	9	27	31	43	138	468	44	760
GENERAL_AGENT		2	19	14	50	49	1	135
THEM_PRONOUN_AGENT	3	2	25	22	56	210	4	322
US_PREVIOUS_AGENT		2			1			3
REF_US_AGENT	59	16	2	8	6	63	6	160
US_AGENT	21	21	40	32	74	959	24	1171
COMPARISON_FORMULAIC		1		9	6	6		22
THEM_AGENT	5	53	16	29	169	86	4	362
—	17	40	364	142	987	4262	21	5833
Total	207	226	720	596	2014	8436	223	12422

Figure 5.20: Contingency Table for Formulaic Expressions Feature (`Formu`)

larly undesirable, as they have knock-on effects on agent determination. An example for such a voice error is the following sentence (underlined; syntactic information about clause-like units is attached to the respective finite verb):

> **At the point where John** <FINITE TENSE="PRESENT" VOICE="ACTIVE" MODAL="NOMODAL" NEGATION="0" ACTIONTYPE="0"> **knows** </FINITE> **the truth** <FINITE TENSE="PRESENT_PERFECT" VOICE="PASSIVE" MODAL="NOMODAL" NEGATION="0" ACTIONTYPE="0"> **has** </FINITE> **been processed, a complete clause**

> <FINITE TENSE="FUTURE_PERFECT" VOICE="ACTIVE" MODAL="NOMODAL" NEGATION="0" ACTIONTYPE="0"> **will** </FINITE> **have been built.** (S-15, 9502035)

This error was caused by the fact that the threading of auxiliaries in our algorithm did not foresee this particular combination of voice and tense. Note that apart from the voice error, everything else is correct. The high level of accuracy achieved in the syntactic processing is not a trivial result, as the processing encompasses complicated combinations of voice, complex tenses and modal auxiliaries, as exemplified by the following corpus example:

> **The actor** <FINITE TENSE="PRESENT_CONTINUOUS" VOICE="ACTIVE" MODAL="NOMODAL" NEGATION="0" ACTIONTYPE="0"> **is** </FINITE> **always running and** <FINITE TENSE="PRESENT" VOICE="ACTIVE" MODAL="NOMODAL" NEGATION="0" ACTIONTYPE="AFFECT"> **decides** </FINITE> **at each iteration whether to speak or not (according to turn-taking conventions); the system** <FINITE TENSE="PRESENT" VOICE="ACTIVE" MODAL="NOMODAL" NEGATION="NEGATED" ACTIONTYPE="NEED"> **does** </FINITE> **not need to wait until a user utterance** <FINITE TENSE="PRESENT" VOICE="PASSIVE" MODAL="NOMODAL" NEGATION="0" ACTIONTYPE="RESEARCH"> **is** </FINITE> **observed to invoke the actor, and** <FINITE TENSE="PRESENT" VOICE="ACTIVE" MODAL="MODAL" NEGATION="NEGATED" ACTIONTYPE="0" > **need** </FINITE> **not respond to user utterances in an utterance by utterance fashion.** (S-137, 9407011)

Contingency tables for features `Syn-1`, `Syn-2` and `Syn-3` can be found in figures 5.21, 5.22 and 5.23, respectively.

It can be the case that more than one finite verb occurs in a sentence, but our main classification method allows only one feature value per feature. All other factors being equal, we prefer verbs in the beginning of the sentence, for two reasons: in the case of coordination, we assume that the more important material might have been presented first; in the case of subordination, we assume that matrix verbs carry more information with respect to meta-discourse. We choose the values associated with the first verb for which `Ag-1` and `Ag-2` returns a non-zero value, or, if not applicable, those for which `Ag-1` returns a non-zero value, or, if not applicable, those for which `Ag-2` returns a non-zero value. Failing all of these alternatives, we chose the values of the first verb in the sentence.

`Syn-1`	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
Active	175	149	407	446	1214	5079	168	7638
Passive	20	62	109	76	363	1286	39	1955
NoVerb	12	15	204	74	437	2071	16	2829
Total	207	226	720	596	2014	8436	223	12422

Figure 5.21: Contingency Table for Voice Feature (`Syn-1`)

`Syn-2`	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
Present Tense	134	158	444	410	1265	5033	177	7621
Present Continuous		4	8	6	18	99	1	136
Past Tense	15	35	23	66	182	819	6	1146
Past Continuous					2	7		9
Past Perfect					1	7		8
Present Perfect	35	10	33	27	88	185	3	381
Future	11	4	8	13	21	211	20	288
Future Continuous						3		3
Future Perfect						1		1
NoVerb	12	15	204	74	437	2071	16	2829
Total	207	226	720	596	2014	8436	223	12422

Figure 5.22: Contingency Table for Tense Feature (`Syn-2`)

`Syn-3`	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
Non_Modal	186	195	422	462	1437	5545	200	8447
Modal	9	16	94	60	140	820	7	1146
NoVerb	12	15	204	74	437	2071	16	2829
Total	207	226	720	596	2014	8436	223	12422

Figure 5.23: Contingency Table for Modal Feature (`Syn-3`)

5.3.3.8. Action Matching

Action Matching determines the value of feature `Ag-2` (contingency table in figure 5.24). It relies on the processing done in the syntactic processing, which determines the *semantic* verb along with the *finite* verb, and also determines whether or not negation was present. Depending on the tense, semantic and finite verb can be the same word. Our algorithm thus performs a distinction between auxiliary and full verb sense for *"have", "be"* and *"do"*. The base form of the semantic verb is determined and it is checked if it is contained in the action lexicon.

Ag-2	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
Positive								
AFFECT		2	5	3	11	68		89
ARGUMENTATION	4	2	2	6	26	62	6	108
AWARE				1	1	2		4
BETTER_SOLUTION	1	1	3	9	5	38		57
CHANGE	4	11	13	11	58	187	5	289
COMPARISON	3	1	2	8	5	50	2	71
CONTINUE	2	21	8	1	20	54		106
CONTRAST		1		5	1	19	1	27
COPULA	24	28	156	112	410	1675	6	2411
FUTURE_INTEREST				1	4	21		26
INTEREST	35	4	27	19	56	209	11	361
NEED		2	19	21	42	186		270
POSSESSION	2	2	25	16	43	204		292
PRESENTATION	78	25	38	39	196	533	105	1014
PROBLEM	1		10	26	18	86	1	142
RESEARCH	11	29	47	38	181	831	17	1154
SIMILAR		10	2	2	8	17		39
SOLUTION	11	16	31	50	135	455	11	709
TEXTSTRUCTURE	1	3	2	3	14	66	27	116
USE	3	22	26	21	98	341	3	514
Negated								
AFFECT			2		1	10		13
ARGUMENTATION			2		2	12		16
AWARE				3		1		4
BETTER_SOLUTION			1	1		1		3
CHANGE			2	3	1	10		16
COMPARISON				2		1		3
CONTINUE			1	3		1		5
CONTRAST						1		1
COPULA	3		18	28	34	209		292
FUTURE_INTEREST						1		1
INTEREST				4	1	18		23
NEED			1	4	5	26		36
POSSESSION			5	3	3	46	1	58
PRESENTATION			3	4	2	17		26
PROBLEM				2	1	8		11
RESEARCH			4	5	3	53		65
SOLUTION			4	13	4	46		67
USE			2	5	2	14		23
0	24	46	259	124	623	2857	27	3960

Figure 5.24: Contingency Table for Action Feature (Ag-2)

If the base form is found in the lexicon, its Action Type is returned; otherwise ActionType 0 is returned (examples for this can be seen in the example sentences on p. 209, where no negation was detected, and where the only two Actions recognized were a (negated) NEED_ACTION—*"the system does not need to wait"* and a (passive) RESEARCH_ACTION—*"a user utterance is observed"*).

In our sample of 100 sentences containing finite verbs, there were no errors introduced in the action type determination step. Appendix B.7 (p. 300) gives an impression of the output of our algorithm on the example article. Recognized actions are shown in light blue boxes; the table on p. 301 gives the corresponding action types.

5.3.3.9. Agent Matching

Agent Matching determines the value of feature `Ag-1` (contingency table in figure 5.25). The algorithm is as follows:

1. Start from the next (initially, the first) finite verb in the sentence;

2. Search for the agent either as a by-PP to the right, or as a subject-NP to the left, depending on the voice associated with the finite verb. The search algorithm tries to stay within the clause that belongs to the finite verb, i.e. it will not cross assumed clause boundaries (e.g. commas or other finite verbs).

3. If one of the Agent Patterns matches within that area in the sentence, return the Agent Pattern and its Agent Type. Else return Agent 0.

4. Repeat Steps 1, 2, 3 until there are no more finite verbs left.

We first evaluated the correctness of the algorithm by randomly taking 100 sentences which contain agent patterns. These 100 sentences contained 111 agents. Apart from erroneous voice determination (cf. section 5.3.3.7), errors could also potentially be introduced by our heuristic for clauses, which never steps over commas and is stopped by appositions, for example.

But in 105 of our sample cases, the agent pattern was syntactically correct: the pattern was matched as prescribed in the pattern, and the matched string agent covered the entire subject of the sentence (active case) or the by-PP with the agent-interpretation (passive case). In 5 of the 111 sentences, the pattern was only *part* of a subject NP (typically the NP in a post-modifying PP), as in the following examples (recognized patterns underlined):

the relations in the models (S-131, 9408014)
the problem with these approaches (S-12, 9504017)

We argue that these cases should not be counted as errors, as they still give an indication of which type of agents the NP should be associated with. In the one sentence with a complete error, this error was due to a mistagging at the POS-Stage (100% precision). No agent pattern that should have been identified was missed (100% recall). Appendix B.7 also shows the output of the agent recognition for the example paper (pink boxes).

`Ag-1`	AIM	BAS	BKG	CTR	OTH	OWN	TXT	Total
US_AGENT	107	85	53	71	114	1456	93	1979
OUR_AIM_AGENT	10			1		5		16
THEM_AGENT		24	9	56	224	59		372
THEM_PRONOUN_AGENT	2		31	24	57	232	1	347
GENERAL_AGENT		1	13	15	28	34	1	92
US_PREVIOUS_AGENT		2		3	37	10		52
REF_AGENT	10	22	20	56	95	374	9	586
REF_US_AGENT	34	3	2	3	1	20	4	67
AIM_REF_AGENT	7		10	1	9	42		69
TEXTSTRUCTURE_AGENT	2	1			4	6	59	72
GAP_AGENT				5		3		8
SOLUTION_AGENT		1	3	5	14	45	3	71
PROBLEM_AGENT			6	2	8	60		76
—	35	87	573	354	1423	6090	53	8615
Total	207	226	720	596	2014	8436	223	12422

Figure 5.25: Contingency Table for Agent Feature (`Ag-1`)

5.3.4. Statistical Classifiers

There are many machine learning algorithms which are able to classify items into predefined categories, given a set of sentential features. *Supervised* methods take information into account which can only be provided externally (the "correct" answer) whereas unsupervised techniques learn without such external provision of the correct answer.

For our task, we use a set of supervised methods because we only have a small set of data (unsupervised methods typically need much more data), and because supervised learning provides the convenient built-in feature of a simple intrinsic evaluation. Also, we follow Kupiec et al. (1995) who have received good results with a simple classifier for the task of determining global sentence relevance (text extraction).

$$P(s \in S|F_1,\ldots,F_k) = \frac{P(F_1,\ldots,F_k|s\in S)P(s\in S)}{P(F_1,\ldots,F_k)} \approx \frac{P(s\in S)\prod_{j=1}^{k} P(F_j|s\in S)}{\prod_{j=1}^{k} P(F_j)}$$

$P(s \in S\|F_1,\ldots,F_k)$:	Probability that sentence s in the source text is included in summary S, given its feature values;
$P(s \in S)$:	Probability that a sentence s in the source text is included in summary S unconditionally; compression rate of the task (constant);
$P(F_j\| s \in S)$:	probability of feature-value pair occurring in a sentence which is in the summary;
$P(F_j)$:	probability that the feature-value pair occurs unconditionally;
k:	number of feature-value pairs;
F_j:	j-th feature-value pair.

Figure 5.26: Kupiec et al.'s (1995) Naive Bayesian Classifier

After having determined a baseline performance with a Naive Bayesian classifier, we then use a more sophisticated method to improve the results of classification. It estimates a better prior probability from the context in terms of the surrounding categories.

5.3.4.1. Naive Bayes

Kupiec et al. were the first to report extraction experiments using a statistical classification method for heuristic combination for determination of global sentence relevance.

Kupiec et al. use the Naive Bayesian Classifier given in figure 5.26. The target value is an estimate of the probability of a sentence to be contained in the abstract, given its feature values. $P(F_j|s \in S)$. In order to estimate this value, probabilities associated with individual events (features) are accumulated; $P(F_j)$ and $P(F_j|s \in S)$ can be estimated from the corpus by raw frequencies. The feature combination applied in a Naive Bayesian model is extremely simple: all conditional probabilities are multiplied.

Kupiec et al. use cross-validation for measuring the success of their classifier: the system extracts sentences from a test document, using a model which was acquired not using any information in the test document. Evaluation can then be measured in precision and recall by the simple criterion of *co-selection* between gold standard and extracted material. Precision gives the percentage of all sentences selected correctly (co-selected with the gold standard) over the total number of sentences selected. Recall gives the percentage of sentences selected correctly (co-selected with the gold standard) over all sentences in the target extract.

In Kupiec et al.'s evaluation, the numerical values for precision and recall are always identical: they use the information of how many gold standard summaries each test document has (though this information would not be available for completely new test documents without abstracts), and their method then extracts the same number of sentences. The method Kupiec et al. chose is a less time consuming way to get an estimation of the cross-over point. (To measure the cross-over point, compression rates are manipulated such that the function of precision and recall can be plotted; the cross-over point of the two functions is then reported.) Another commonly accepted combination of precision and recall is F-measure (van Rijsbergen, 1979).

In (Teufel and Moens, 1997), we report a duplication of Kupiec et al.'s experiment for text extraction. With different data and two types of gold standards, but with similar features to Kupiec et al., we achieved favourably comparable results (cf. the left two columns in figure 5.27). In Kupiec et al.'s case, the best precision and recall of 44% was reached by combining location, cue phrase and sentence length features; in ours, the best result of 68% was achieved using all five features.

	Kupiec et al.		Our replication	
Heuristics	Individual	Cumulative	Individual	Cumulative
Cue Phrases	33%	33%	55%	55%
Location	29%	42%	32%	65%
Sentence Length	24%	44%	29%	66%
tf/idf	20%	42%	17%	67%
Capitalization + *tf/idf*	20%	42%	—	
Title	—		21%	68%
Baseline	24%		28%	

Figure 5.27: Results of our Duplication of Kupiec et al.'s (1995) experiment

But here we adapt Kupiec et al.'s Naive Bayesian formula (figure 5.26) for Argumentative Zoning, resulting in the formula given in figure 5.28. As far as the notation is concerned, let us assume we have n features F_0 to F_{n-1}; a feature is then known as F_j, with $0 \leq j < n$. Each of the features F_j has k_j different values V_{jr}, with $0 \leq r < k_j$. There are m target categories C^0 to C^{m-1}; a target category is then known as C^i, with $0 \leq i < m$. In our case, m is 7 (whereas Kupiec et al. perform binary classification; $m = 2$), n is 16, and the k_j vary from 2 for j= 0,1,6 (`Cont-1, Cont-2, Length`) to 40 for j=15 (`Ag-2`).

F_4=Struct-2	C^0= AIM	C^1= BAS	C^2= BKG	C^3= CTR	C^4= OTH	C^5= OWN	C^6= TXT	Total
$V_{4,0}$=Initial	$n^0_{4,0}$= 117	$n^1_{4,0}$= 92	$n^2_{4,0}$= 267	$n^3_{4,0}$= 135	$n^4_{4,0}$= 601	$n^5_{4,0}$= 2532	$n^6_{4,0}$= 73	$n_{4,0}$= 3817
$V_{4,1}$=Medial	$n^0_{4,1}$= 56	$n^1_{4,1}$= 87	$n^2_{4,1}$= 306	$n^3_{4,1}$= 289	$n^4_{4,1}$= 971	$n^5_{4,1}$= 3779	$n^6_{4,1}$= 68	$n_{4,1}$= 5556
$V_{4,2}$=Final	$n^0_{4,2}$= 34	$n^1_{4,2}$= 47	$n^2_{4,2}$= 147	$n^3_{4,2}$= 172	$n^4_{4,2}$= 442	$n^5_{4,2}$= 2125	$n^6_{4,2}$= 82	$n_{4,2}$= 3049
Total	n^0= 207	n^1= 226	n^2= 720	n^3= 596	n^4= 2014	n^5= 8436	n^6= 223	N= 12422

Figure 5.29: Contingency Table for Paragraph Feature

$$P(C^i|V_{0,x},...,V_{n-1,y}) = P(C^i)\frac{P(V_{0,x},...,V_{n-1,y}|C^i)}{P(V_{0,x},...,V_{n-1,y})} \approx P(C^i)\frac{\prod_{j=0}^{n-1}P(V_{j,r}|C^i)}{\prod_{j=0}^{n-1}P(V_{j,r})}$$

$P(C^i\|V_{0,x},\ldots,V_{n-1,y})$:	Probability that a sentence has target category C^i, given its feature values $V_{0,x}, \ldots, V_{n-1,y}$, with $0 \leq x < k_0$ and $0 \leq y < k_{n-1}$;
$P(C^i)$:	Probability that a sentence has target category C^i (prior);
$P(V_{j,r}\|C^i)$:	Probability of feature-value pair $V_{j,r}$ occurring with target category C^i;
$P(V_{j,r})$:	Probability of feature value $V_{j,r}$ (rth value of Feature F_j);

Figure 5.28: Our Adaptation of Kupiec et al.'s (1995) Naive Bayesian Classifier

The first part of the second formula, $P(C^i)$, is called the *prior* probability, and the second part $\frac{P(V_{0,x},...,V_{n-1,y}|C^i)}{P(V_{0,x},...,V_{n-1,y})}$ is called the *posterior* probability. The first derivation is due to Bayes' Theorem; the second is specific to the Naive Bayesian formula and only legal under the Independence Assumption, i.e. the assumption that all features are statistically independent $(P(F_1,F_2) = P(F_1) * P(F_2))$. If, however, the data show that certain features are statistically dependent on each other—and to a certain degree this can be expected, as it is difficult to define features that are statistically independent—the Naive Bayes method will not result in an absolutely accurate language model.

We will now describe how the conditional probability $P(V_{j,r}|C_i)$ needed for Naive Bayesian classification can be calculated from the contingency tables.

For example, in figure 5.29 (repeated from figure 5.4), the vertical totals $n_{j,r}$

give the occurrence counts of feature value $V_{j,r}$ ($n_{j,r}$ is a short notation for frequency $f(V_{j,r})$); the horizontal totals n^i (or $f(C^i)$) give the occurrence counts of category C_i, and the data cells $n^i_{j,r}$ (or $f(Vj,r,C^i)$) give the number of occurrences of category C_i with feature value V^r_j. N is the number of all items.

Then the desired probability $P(V_{4,1}|C^0)$, i.e. the probability that a sentence displays the feature value $V_{4,1}$ (`Medial`) of feature `Struct-2` , given that the target class of the sentence is AIM, with $i = 0$, $j = 4$ and $r = 1$ (C^0=`Aim`; F_4 = `Struct-2`; and $V_{4,1}$=`Medial`), can be estimated by corpus frequencies $f(Vj,r,C^i)$ and $f(C^i)$ as follows:

$$P(V_{jr}|C^i) = \frac{f(Vj,r,C^i)}{f(C^i)} = \frac{|n^i_{j,r}|}{|n^i|}$$

$$P(Medial|Aim) = P(V_{4,1}|C^0) = \frac{|n^0_{4,1}|}{|n^0|} = \frac{56}{207} = 0.27.$$

It is obvious that for each category C^i and for each feature F_j, the following equality holds:

$$\sum_{r=0}^{k_j-1} P(V_{j,r}|C^i) = 1$$

Naive Bayes estimates the prior probability $P(C^i)$ by simple unigram frequency:

$$P(C^i) = \frac{|n^i|}{|N|}$$

$$P(Aim) = \frac{207}{12422} = 0.0166$$

The reverse probability is $P(C^i|V_{j,r})$: the probability that, on the basis of a given observed feature $V_{j,r}$, the sentence will be classified as C^i. This probability is not used in our calculation.

Naive Bayes estimates the posterior under the independence assumption, but we suspect that our features are not really independent. Intuitively it is clear that they must be related to each other: certain agents, for example GENERAL_AGENT, tend to occur more often in initial locations in the document. This interaction is highly relevant for our experiment. However, it is less obvious which of the features (if any) is directly related to sentence length. A more sophisticated classifier for the posterior probability

$\frac{P(V_{0,x},\ldots,V_{n-1,y}|C^i)}{P(V_{0,x},\ldots,V_{n-1,y})}$ does not simply derive the posterior by multiplication of the single probabilities; it determines which features are independent and only multiplies their conditional probabilities. Because of this, we expect better classification results for more sophisticated classifiers. We use two such algorithms, the rule-learning classifier RIPPER (Cohen, 1995, 1996) and a Maximum Entropy-based classifier (Mikheev, To Appear).

5.3.4.2. N-Gram Modelling

In Naive Bayes, not only the posterior, but also the prior is estimated in a very simple manner: it is constant all over the document. However, our model of the typical flow of argumentation predicts typical patterns in our texts. We know that a sentence is more likely to be of category AIM, for example, if the previous sentence was a CONTRAST (introducing a gap), than if the previous sentence was an OTHER sentence (neutrally describing other work)—even if we do not know anything about the features of the sentence to be classified yet. The simple Bayesian classifier, however, does not exploit this fact, i.e. it does not use the context.

N-gram models estimate a more accurate prior by taking the context of a sentence, in terms of surrounding categories, into account. N-gram models are typically used over letters in statistical language processing, but we apply them to *whole sentences* instead. The prior can then be written as $P(C^i_m|C_{m-1},\ldots,C_{m-o})$, for the m-th sentence in the document, instead of $P(C^i)$. The index $o+1$ is called the *order* of the ngram model. A system of order $o+1$ takes o items before the one to be classified into account—a bigram model ($o+1=2$) uses the formula $P(C^i_m|C_{m-1})$.

We ran experiments with N-gram models of order 2, 3 and 4 to estimate the priors, after we first determined the posterior probabilities with the Naive Bayesian model.

$$P(C^i_m|V_{0,x},\ldots,V_{n-1,y}) \approx \; P(C^i_m|C_{m-1},\ldots,C_{m-o}) \; P(C^i)\frac{\prod_{j=0}^{n-1}P(V_{j,r}|C^i)}{\prod_{j=0}^{n-1}P(V_{j,r})}$$

For parameter estimation, we use the Edinburgh Speech Tools Library (Taylor et al., 1999), which use the Viterbi algorithm to maximize the prior probabilities.

5.3.5. Symbolic Rules

We have provided a set of symbolic rules for the determination of the four non-basic categories AIM, TEXTUAL, BASIS and CONTRAST. The rules rely on the sentential features (mainly the Agentivity features), and provide a high-precision, low-recall extraction. For many applications, precision is more important than recall: few sentences might be sufficient, provided that they can be determined with a high level of confidence.

The first step in the algorithm is to assign each sentence scores for each of the categories, whereby several factors are taken into account. These scores are assigned by symbolic rules. Figures 5.30 and 5.31 give the rules for AIM scores. We use two different algorithms for choosing sentences: Method I takes all sentences whose score is above threshold, whereas Method II only takes two sentences who are above threshold: one in the beginning, and one in the end (i.e., one from the introduction and one from the conclusions). Method II is only used for AIM sentences.

We empirically established good threshold values for the scores assigned in the symbolic processing. Figure 5.32 shows how the thresholds relate to precision and recall values achieved with both algorithms on AIM sentences. For high thresholds, Method II achieves a very high precision, albeit a little lower recall than Method I. This might be the method of choice for determining AIM sentences with a high level of certainty. For example, with Method II, the score of 11 gives us a 96% precision and a 23% recall. For lower thresholds (this might be good for determining "second best" candidates), Method I is advantageous, as Method II cannot achieve recall higher than 48% in our case (not all AIM sentences occur in the beginning and end of a document, and some documents contain more than two AIM sentences).

5.4. Intrinsic Evaluation

Evaluation of the systems relies on 10-fold cross-validation: the model is trained on a training set of 72 documents, leaving 8 documents out at a time (the test set). The model is then used on the test set to assign each sentence a probability for each category R, and the category with the highest probability is chosen as answer for the sentence. This is repeated for all ten folds. The baselines for this task were discussed in section 4.2.

5.4.1. Naive Bayes Model

As Naive Bayes does not automatically ignore useless features, and as performance with bad features decreases, the first question is if all of our features are good disambiguators, or if some of the features do not contribute any useful information. Figure 5.33 shows the results of a 10-fold cross-validation.

Condition	Score
Start	Score = 0
If sentence in beginning	Score + 1
If sentence not in beginning	Score – 1
If Ag-1 = OUR_AIM_AGENT and Ag-2 = COPULA (non-negated) and first action in sentence and beginning (i.e. Loc = A, B, C, D or E	Score = 8
If Ag-1 = OUR_AIM_AGENT and Ag-2 = COPULA (non-negated) and first action in sentence and not beginning	Score = 6
If Ag-1 = OUR_AIM_AGENT and Ag-2 = COPULA (non-negated) and not first action in sentence and beginning	Score = 6
If Ag-1 = OUR_AIM_AGENT and Ag-2 = COPULA (non-negated) and not first action in sentence and not beginning	Score = 4
If Ag-1 = US_AGENT and Ag-2 = PRESENTATION_ACTION (non-negated) and first action in sentence and beginning	Score = 6
If Ag-1 = US_AGENT and Ag-2 = PRESENTATION_ACTION (non-negated) and first action in sentence and not beginning	Score = 4
If Ag-1 = US_AGENT and Ag-2 = PRESENTATION_ACTION (non-negated) and not first action in sentence and beginning	Score = 4
If Ag-1 = US_AGENT and Ag-2 = PRESENTATION_ACTION (non-negated) and not first action in sentence and not beginning	Score = 2
If Ag-1 = US_AGENT and Ag-2 = INTEREST_ACTION (non-negated) and first action in sentence and beginning	Score = 5
If Ag-1 = US_AGENT and Ag-2 = INTEREST_ACTION (non-negated) and first action in sentence and not beginning	Score = 3
If Ag-1 = US_AGENT and Ag-2 = INTEREST_ACTION (non-negated) and not first action in sentence and beginning	Score = 3
If Ag-1 = US_AGENT and Ag-2 = INTEREST_ACTION (non-negated) and not first action in sentence and not beginning	Score = 1
If Ag-1 = (REF_)US_AGENT and Ag-2 = SOLUTION_ACTION (non-negated) and first action in sentence and beginning	Score = 3
If Ag-1 = (REF_)US_AGENT and Ag-2 = SOLUTION_ACTION (non-negated) and first action in sentence and not beginning	Score = 2
If Ag-1 = (REF_)US_AGENT and Ag-2 = SOLUTION_ACTION (non-negated) and not first action in sentence and beginning	Score = 1
If Ag-1 = (REF_)US_AGENT and Ag-2 = SOLUTION_ACTION (non-negated) and not first action in sentence and not beginning	Score = 0
If Ag-1 = (REF_)US_AGENT and Ag-2 = ARGUMENTATION_ACTION (non-negated) and first action in sentence	Score = 3
If Ag-1 = (REF_)US_AGENT and Ag-2 = ARGUMENTATION_ACTION (non-negated) and not first action in sentence	Score = 2
If Ag-1 = REF_AGENT and Ag-2 = INTEREST_ACTION (non-negated) and first action in sentence	Score = 4
If Ag-1 = REF_AGENT and Ag-2 = INTEREST_ACTION (non-negated) and first action in sentence	Score = 3
If Ag-1 = REF_AGENT and Ag-2 = PRESENTATION_ACTION (non-negated) and first action in sentence	Score = 3
If Ag-1 = REF_AGENT and Ag-2 = PRESENTATION_ACTION (non-negated) and not first action in sentence	Score = 2

Figure 5.30: Symbolic Scores for AIM Sentences (1 of 2)

Condition	Score
If `Ag-1` = AIM_REF_AGENT and `Ag-2` = COPULA (non-negated) and first action in sentence	Score = 4
If `Ag-1` = AIM_REF_AGENT and `Ag-2` = COPULA (non-negated) and not first action in sentence	Score = 3
If `Ag-1` = (REF_)US_AGENT and `Ag-2` = RESEARCH_ACTION (non-negated)	Score = 1
If `Formu` = HERE_FORMULAIC and beginning	Score + 5
If `Formu` = METHOD_FORMULAIC and `Ag-2` = (PRESENTATION_ACTION or INTEREST_ACTION) and `Ag-1` = (REF_US_AGENT or REF_AGENT or *AIM*_AGENT)	Score + 5
If `Struct-3` = Introduction	Score + 2
If `Struct-3` = Conclusion	Score + 2
If `Struct-1` = First-sentence	Score + 2
If very first sentence in document	Score + 1
If the previous sentence contained contrastive material (GAP, PROBLEM_ACTION, AWARE_ACTION, CONTRAST_FORMULAIC, negated SOLUTION ACTION), and beginning	Score + 2
If `Ag-1` = US_AGENT	Score + 1
If there was a textstructure sentence in the past 3 sentences	Score – 1
If there is a DETAIL_FORMULAIC in the sentence	Score – 1
If `Ag-1` = REF(_US?)_AGENT and `Ag-2` = TEXTSTRUCTURE_ACTION	Score – 2
If last sentence was classified as TEXTUAL	Score – 3
If `Ag-1` = (ref_)?us_agent and `Ag-2` = PRESENTATION_ACTION and `Syn-2` = Present and not beginning	Score – 2
If `Ag-1` = TEXTSTRUCTURE_AGENT and `Ag-2` = (TEXTSTRUCTURE_ACTION or PRESENTATION_ACTION or INTEREST_ACTION or RESEARCH_ACTION) or `Formu` = TEXTSTRUCTURE_FORMULAIC or `formu` = TEXTSTRUCTURE_AGENT	Score = 0
If there is a US_PREVIOUS_FORMULAIC in the sentence	Score = 0
If there is a FUTURE_FORMULAIC in the sentence	Score = 0

Figure 5.31: Symbolic Scores for AIM Sentences (2 of 2)

Feature	Alone	Left out	Feature	Alone	Left out
`Cont-1`	K=–.12	.37	`Syn-2`	K=–.12	.37
`Cont-2`	K=–.12	.37	`Syn-3`	K=–.12	.37
`Struct-1`	K=–.12	.36	`Cit-1`	K=+.18	.38
`Struct-2`	K=–.12	.37	`Cit-2`	K=+.13	.38
`Struct-3`	K=+.05	.35	`Cit-3`	K=+.12	.38
`Loc`	K=+.17	.34	`Formu`	K=+.06	.35
`Length`	K=–.12	.37	`Ag-1`	K=+.07	.36
`Syn-1`	K=–.12	.37	`Ag-2`	K=–.11	.35

Figure 5.33: Performance of Individual Features (Naive Bayes)

The first column in figure 5.33 (“Alone”) corresponds to classification with a model using only the given feature, whereas the second column (“Left out”) corresponds to a model using all other features but the given one. Some of the weaker

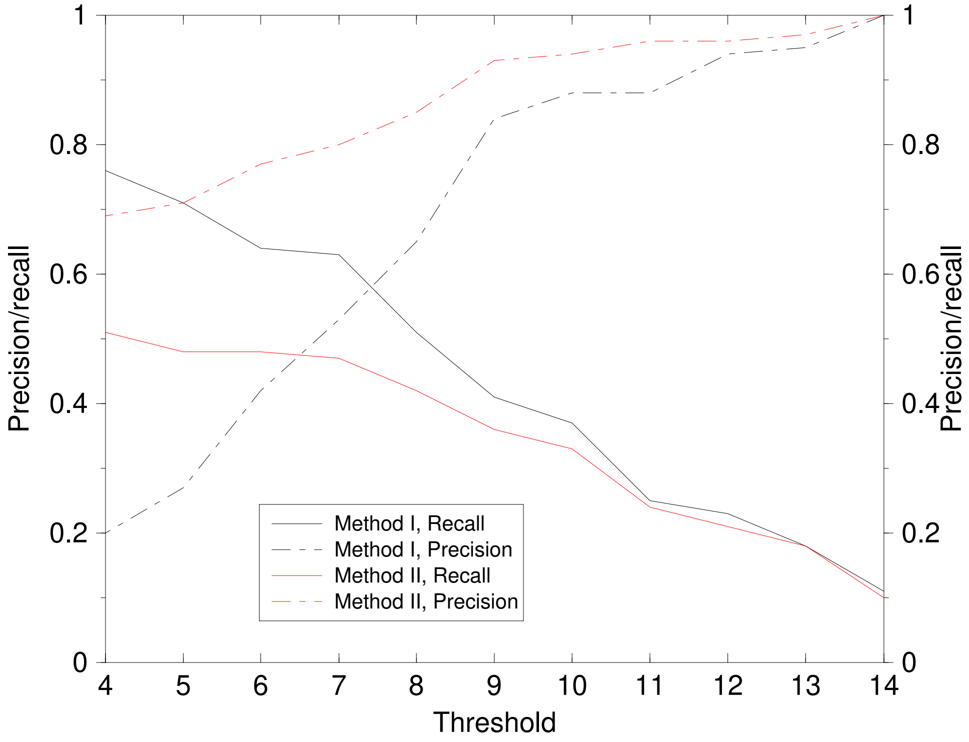

Figure 5.32: Effect of Threshold on Symbolic AIM Sentence Extraction

features are not predictive enough on their own to break the dominance of the prior; in that case, they behave just like Baseline B1 (K=–.12). A distinctive feature has a good classification on its own, and leads to a decreased performance if left out. The numbers show that some of the weaker features contribute some predictive power in combination with others, even if not on their own.

We measured the best performance using the features `Cont-1`, `Cont-2`, `Loc`, `Struct-1`, `Struct-2`, `Struct-3`, `Length`, `Syn-1`, `Syn-2`, `Syn-3`, `Cit-1`, `Formu`, `Ag-1` and `Ag-2`. Results only decreased when combinations of the citation features were used together; we assume this is due to the fact that these features encode redundant information with respect to each other; they are not independent. Appendix B.8 shows the output of the Naive Bayesian model on the example paper. The system's annotation achieved a Kappa value of K=0.41 on the example paper.

In an experiment between one annotator (C) and the statistical method, the observed reproducibility is K=.39 (N=12421, k=2), which corresponds to percentage

		MACHINE (NAIVE BAYES)							
		AIM	CTR	TXT	OWN	BKG	BAS	OTH	Total
HUMAN	AIM	131	8	11	33	14	7	5	209
	CTR	22	124	2	259	80	24	86	597
	TXT	13	3	138	51	6	5	7	223
	OWN	116	116	62	7623	163	96	257	8433
	BKG	28	40	3	257	305	11	76	720
	BAS	14	9	4	48	5	91	56	227
	OTH	8	71	10	1115	198	122	489	2013
	Total	332	371	230	9386	771	356	976	12422

Figure 5.34: Confusion Matrix: Human vs. Automatic Annotation, Naive Bayes

accuracy of 71.2%.

Note here that the system is not asked to annotate abstract sentences, so that N is lower than it would have been in a comparable experiment involving only human annotators. This number cannot be directly compared to experiments like Kupiec et al.'s because in their experiment a compression of around 3% was achieved whereas we classify each sentence into one of the categories.

When the Naive Bayesian Model is added to the pool of 3 coders, the reproducibility drops from K=.71 to K=.54 (N=3446, n=4). This reproducibility value is equivalent to the value achieved by 6 human annotators with no prior training, as in Study III.

Figure 5.34 depicts the confusion matrix for the classification. We can see that the system guesses too few OTHER and CONTRAST sentences, but overestimates the

	AIM	CTR	TXT	OWN	BKG	BAS	OTH
Precision	39%	33%	60%	81%	40%	26%	50%
Recall	63%	21%	62%	91%	42%	40%	24%

Figure 5.35: Precision and Recall per Category, Naive Bayes

number of BASIS sentences.

Figure 5.35 shows that the system performs well on AIM sentences, which can be determined with a recall of 63% and a precision of 39%. These values are more directly comparable to Kupiec et al.'s results of 44% precision and 44% recall for extracted sentences, even though not all of the sentences extracted by their method would have fallen into our AIM category. The other easily determinable category for the automatic method is TEXTUAL (p=60%; r=62%), whereas the results for the other non-basic categories are relatively lower—as are the human annotation results.

The results achieved with the more complicated statistical techniques were not much better. RIPPER (Cohen, 1995, 1996) achieved an error rate of 27.66% +/- 0.35% (a bit better than our error rate of 29%) in a ten-fold cross-validation. When the classifier described in Mikheev (To Appear) was used on our data, the classification was minimally better than both the Naive Bayes model and RIPPER, but training this model is very time consuming.

5.4.2. N-Gram Model

We measured performance of different n-gram models as before by 10-fold cross-validation. The best performance was achieved with a bigram model. This model achieved K=.41 (n=2,N=12422) when compared to Annotator C alone (P(A)=0.703, P(E)=0.492), and K=.56 (N=3334, n=4, P(A)=0.795, P(E)=0.537) when added to the pool of three annotators. Thus, adding the bigram model does improve performance. Appendix B.9 (p. 303) shows the output of the bigram model on the example paper. If we compare it to the output of the Naive Bayes model (p. 302), we notice that the contextual information introduced by the bigram model has added useful aspects to the annotation. For example, the Naive Bayes model did not annotate the two sentences dealing with Hindle's approach (bottom of the first column) as either OTHER or CONTRAST; instead, it just left them as BACKGROUND. Because of the high probability of CONTRAST sentences preceding AIM sentences, the Viterbi algorithm chose to mark

		MACHINE (BIGRAM)							
		AIM	CTR	TXT	OWN	BKG	BAS	OTH	Total
	AIM	124	10	12	27	25	3	8	209
	CTR	20	122	3	208	138	15	91	597
	TXT	13	4	133	51	11	3	8	223
HUMAN	OWN	107	138	68	7220	459	99	342	8433
	BKG	9	20	3	141	454	5	88	720
	BAS	18	14	4	69	12	80	30	227
	OTH	3	97	7	797	395	117	597	2013
	Total	294	405	230	8513	1494	322	1164	12422

Figure 5.36: Confusion Matrix: Human vs. Automatic Annotation, Bigram Model

them as CONTRAST; the fact that the posterior probability for CONTRAST was slightly lower than the posterior probability for AIM was overridden by the prior probabilities. Similarly, the erroneously tagged TEXTUAL sentence at the end of the introduction is corrected by the bigram model into CONTRAST.

In general, the bigram model tends to annotate longer segments; posterior probabilities have to be high to break this preference, i.e., to start new segments. This also introduces errors, e.g., the long CONTRAST segment at the end of the second column which was not perceived to be there by either human annotator. Overall, the bigram model's annotation reached a Kappa value of 0.35 on this particular paper, i.e. performance *decreased* when compared to the Naive Bayesian model.

For the case of human vs. bigram model, the confusion matrix in figure 5.36 was recorded. Figure 5.37 shows precision and recall values for individual categories. In contrast to the Naive Bayesian model, the recognition results for the categories AIM,

	AIM	CTR	TXT	OWN	BKG	BAS	OTH
Precision	42%	30%	58%	85%	30%	25%	51%
Recall	59%	20%	60%	86%	63%	35%	30%

Figure 5.37: Precision and Recall per Category, Bigram Model

OTHER and OWN are higher, and those for the categories CONTRAST, TEXTUAL, BASIS and BACKGROUND lower.

5.4.3. Symbolic Rules

The symbolic rules do not aim at a full-coverage recognition of all categories. Rather, they provide a high-precision, low-recall coverage of the four non-basic categories AIM, TEXTUAL, BASIS and CONTRAST. The evaluation of the success of these rules can therefore not be measured by Kappa (which would require a full-coverage classification), but only by precision and recall of these four categories. Precision and recall was varied by changing the threshold.

Figure 5.38 presents precision and recall plots for the non-basic categories. The results show that it is possible to determine AIM and TEXTUAL sentences in a scientific article with high precision, albeit with considerably lower recall. This is a good result, which in itself justifies the Agentivity features. The result is also in agreement with our results from chapter 4 which showed that AIM sentences (and to a lesser degree TEXTUAL sentences) are also recognized most robustly of all categories by humans. They state knowledge claims—it is important for authors to bring the own knowledge claims across—or organize the text. Typically, they are expressed in a formalized way. BASIS and CONTRAST sentences have a less prototypical syntactic realization, and they also occur at less predictable places in the document. Therefore, it is far more difficult for both machine and human to recognize such sentences.

Figure 5.38 also shows the best stochastic results for the non-basic categories (dots) for comparison. The results for AIM and CONTRAST are better with the symbolic system, whereas the reverse is the case for the categories BASIS and TEXTUAL.

5.5. Results of System Run on Unseen Material

An ad-hoc test was performed on a paper randomly drawn from the archive. It was preprocessed with minimal manual intervention and then put through the argumentative

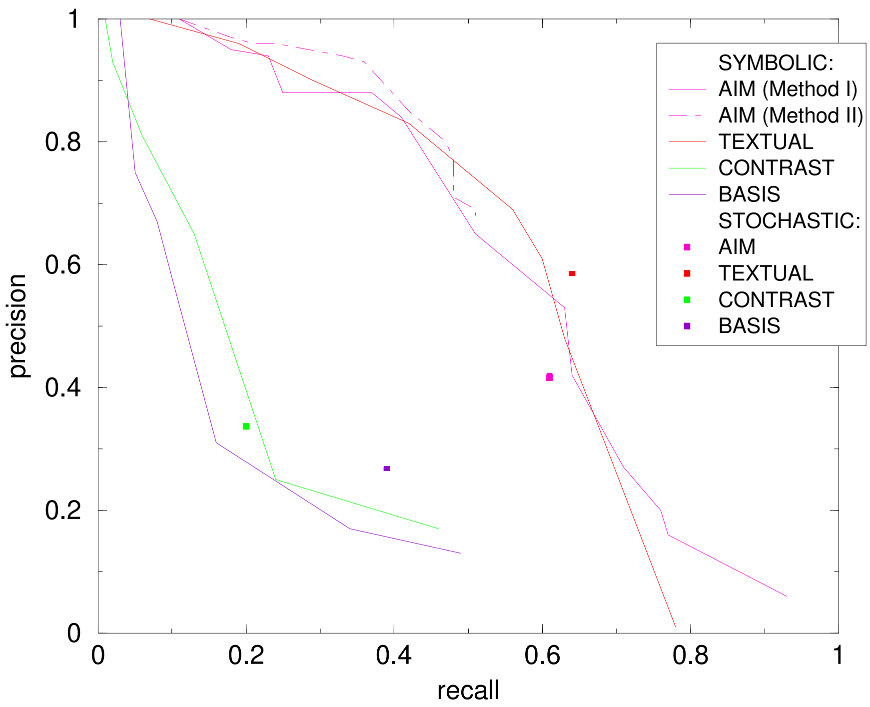

Figure 5.38: Precision and Recall of Symbolic Sentence Extraction

zoner. The output of the Naive Bayesian model is given in figures 5.39 and 5.40, and the output of the bigram model is given in figures 5.41 and 5.42 so that the reader can inspect the result.

The only difference in performance which can be expected when moving from seen to unseen text has to do with the features based on meta-discourse (`Formu`, `Ag-1` and `Ag-2`), as the list of expressions was expanded manually during system development, whenever the system's results showed phrases not previously contained in the lists. All other features are rather independent of the question whether or not the system developer sees more data. One would hope that the common meta-discourse phrases are covered by the list, and that expressions not encountered in the first 80 papers would be rather specialized and infrequent.

It is difficult to assess to what extent our features treat unseen text adequately, because there are no gold standards for the unseen test. We report an experiment with a predecessor of the three meta-discourse features in Teufel and Moens (1997). We divided our corpus (then 123 articles, including articles which did not appear in ACL, EACL, COLING or ANLP conferences) into three parts. We pretended that one third was "unseen", by using only those 1423 formulaic expressions for extraction which

A Simple Transformation for Offline-Parsable Grammars and its Termination Properties

Marc Dymetman -- 9605023 -- Coling 94

Abstract

A-0 We present, in easily reproducible terms , a simple transformation for offline-parsable grammars which results in a provably terminating parsing program directly top-down interpretable in Prolog . A-1 The transformation consists in two steps : A-2 removal of empty productions , followed by : A-3 left-recursion elimination . A-4 It is related both to left-corner parsing (where the grammar is compiled , rather than interpreted through a parsing program , and with the advantage of guaranteed termination in the presence of empty productions) and to the Generalized Greibach Normal Form for DCGs (with the advantage of implementation simplicity) .

Motivation

S-0 Definite clause grammars (DCGs) are one of the simplest and most widely used unification grammar formalisms. S-1 They represent a direct augmentation of context-free grammars through the use of (term) unification (a fact that tends to be masked by their usual presentation based on the programming language Prolog) . S-2 It is obviously important to ask whether certain usual methods and algorithms pertaining to CFGs can be adapted to DCGs , and this general question informs much of the work concerning DCGs , as well as more complex unification grammar formulisms (to cite only a few areas : Earley parsing , LR parsing , left-corner parsing , Greibach Normal Form) .

S-3 One essential complication when trying to generalize CFG methods to the DCG domain lies in the fact that , whereas the parsing problem for CFGs is decidable , the corresponding problem for DCGs is in general undecidable . S-4 This can be shown easily as a consequence of the noteworthy fact that any definite clause program can be viewed as a definite clause grammar " on the empty string " , that is , as a DCG where no terminals other than <EQN/> are allowed on the right-hand side of rules . S-5 The Turing - completeness of definite clause programs therefore implies the undecidability of the parsing problem for this subclass of DCGs , and a fortiori for DCGs in general . S-6 In order to guarantee good computational properties for DCGs , it is then necessary to impose certain restrictions on their form such as offline - parsability (OP) , a nomenclature introduced by Pereira and Warren 1983 , who define an OP DCG as a grammar whose context-free skeleton CFG is not infinitely ambiguous , and show that OP DCGs lead to decidable parsing problem .

S-7 Our aim in this paper is to propose a simple transformation for an arbitrary OP DCG putting it into a form which leads to the completeness of the direct top-down interpretation by the standard Prolog interpreter : parsing is guaranteed to enumerate all solutions to the parsing problem and terminate . S-8 The existence of such a transformation is known : in Dymetman 1992a , Dymetman 1992b , we have recently introduced a " Generalized Greibach Normal Form " (GGNF) for DCGs, which leads to termination of top-down interpretation in the OP case. S-9 However , the available presentation of the GGNF transformation is rather complex (it involves an algebraic study of the fixpoints of certain equational systems representing grammars .) . S-10 Our aim here is to present a related , but much simpler , transformation , which from a theoretical viewpoint performs somewhat less than the GGNF transformation (it involves some encoding of the initial DCG , which the GGNF does not , and it only handles offline-parsable grammar , while the GGNF is defined for arbitrary DCGs) , but in practice is extremely easy to implement and displays a comparable behaviour when parsing with an OP grammar .

S-11 The transformation consists of two steps : S-12 empty-production elimination and S-13 left-recursion elimination .

S-14 The empty-production elimination algorithm is inspired by the usual procedure for context-free grammars . S-15 But there are some notable differences , due to the fact that removal of empty-productions is in general impossible for non-OP DCGs. S-16 The empty-production elimination algorithm is guaranteed to terminate only in the OP case. S-17 It produces a DCG declaratively equivalent to the original grammar .

S-18 The left-recursion elimination algorithm is adapted from a tranformation proposed in Dymetman et al. 1990 in the context of a certain formalism (" Lexical Grammars ") which we presented as a possible basis for building reversible grammars . S-19 The key observation (in slightly different terms) was that , in a DCG , if a nonterminal g is defined literally by the two rules (the first of which is left-recursive) :

[IMAGE]

S-20 then the replacement of these two rules by the three rules (where <EQN/> is a new nonterminal symbol , which represents a kind of " transitive closure " of d) :

[IMAGE]

S-21 presents the declarative semantics of the grammar .

S-22 We remarked in Dymetman et al. 1990 that this transformation is closely related to left-corner parsing " , but did not give details . S-23 In a recent paper Johnson forthcoming introduces " a left-corner program transformation for natural language parsing " , which has some similarity to the above transformations , but which is applied to definite clause grammars, rather than DCGs . S-24 He proves that this transformation respects declarative equivalence , and also shows , using a model -theoretic approach , the close connection of his transformation with left-corner parsing Rosenkrantz and Lewis 1970 , Matsumoto et al. 1983 , Pereira and Shieber 1987 .

S-25 It must be noted that the left-recursion elimination procedure can be applied to any DCG ,whether OP or not. S-26 Even in the case where the grammar is OP , however , it will not lead to a terminating parsing algorithm unless empty productions have been prealably eliminated from the grammar , a problem which is shared by the usual left-corner parser-interpreter .

S-27 Due to the space available , we do not give here correctness proofs for the algorithm presented , but expect to publish them in a fuller version of this paper . S-28 These algorithms have actually been implemented in a slightly extended version , where they are also used to decide whether the grammar proposed for transformation is in fact offline-parsable or not .

Figure 5.39: Unseen Document 9605023, Automatic Argumentative Zoning by Naive Bayes (1 of 2)

Empty-production elimination

S-29 It can be proven that , if DCG0 is an OP DCG , the following transformation , which involves repeated partial evaluation of rules that rewrite into the empty string , terminates after a finite number of steps and produces a grammar DCG without empty-productions which is equivalent to the initial grammar on non-empty strings :

[IMAGE]

S-30 For instance the grammar consisting in the nine rules appearing above the separation in fig. <CREF/> is transformed into the grammar (see figure) :

[IMAGE]

Left-recursion elimination

S-31 The transformation can be logically divided into two steps: S-32 an encoding of DCG into a "generic " form DCG' , and S-33 a simple replacement of a certain group of left-recursive rules in DCG' by a certain equivalent non left-recursive group of rules , yielding a top-down interpretable DCG'' . S-34 An example of the transformation <EQN/> is given in fig. <CREF/> .

S-35 The encoding is performed by the following algorithm :

{IMAGE}

S-36 The procedure is very simple . S-37 It involves the creation of a generic nonterminal g(X) , of arity one , which performs a task equivalent to the original nonterminals <EQN/> . S-38 The goal <EQN/> , for instance , plays the same role for parsing a sentence as did the goal <EQN/> in the original grammar .

S-39 Two further generic nonterminals are introduced : t(X) accounts for rules whose right-hand side begins with a terminal , while d(Y,X) accounts for rules whose right-hand side begins with a non-terminal. S-40 The rationale behind the encoding is best understood from the following examples , where <EQN/> represents rule rewriting :

[IMAGE]

S-41 The second example illustrates the role played by d(Y, X) in the encoding. S-42 This nonterminal has the following interpretation : X is an " immediate " extension of Y using the given rule . S-43 In other words , Y corresponds to an " immediate left corner " of X .

S-44 The left-recursion elimination is now performed by the following " algorithm " :

[IMAGE]

S-45 In this transformation , the new nonterminal <EQN/> plays the role of a kind of transitive closure of d . S-46 It can be seen that , relative to DCG'' , for any string w and for any ground term z , the fact that g(z) rewrites into w -- or , equivalently , that there exists a ground term x such that <EQN/> rewrites into w -- is equivalent to the existence of a sequence of ground terms <EQN/> and a sequence of strings <EQN./> such that t(x1) rewrites to w1, d(x1, x2) rewrites into w2, ... , d(xk -1, xk) rewrites into wk , and such that w is the string concatenation <EQN/> . S-47 From our previous remark on the meaning of d(Y, X) , this can be interpreted as saying that " constituent x is a left-corner of constituent z " , relatively to string w .

S-48 The grammar DCG'' can now be compiled in the standard way -- via the adjunction of two " differential list " arguments -- into a Prolog program which can be executed directly. S-49 If we started from an offline-parsable grammar DCG0 , this program will enumerate all solutions to the parsing problem and terminate after a finite number of steps .

References

Marc Dymetman . A Generalized Greibach Normal Form for Definite Clause Grammars . In proceedings of the 15th International Conference on Computational Linguistics, volume 1, pages 366-372, Nantes, France, July 1992 .

Marc Dymetman. Transformations de grammaires logiques et reversibilites en Traduction Automatique . These d'Etat , 1992 . Universite Joseph Fourier (Grenoble 1) , Grenoble , France .

Marc Dymetman and Pierre Isabelle . Reversible logic grammars for machine translation . In Proceedings of the Second International Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages, Pittsburgh, PA, June 1988. Carnegie Mellon University.

Marc Dymetman, Pierre Isabelle, and Francois Perrault. A symmetrical approach to parsing and generation. In Proceedings of the 13th International Conference on Computational Linguistics, volume 3, pages 90-96, Helsinki , August 1990 .

Andrew Haas. A generalization of the offline-parsable grammars. In Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics, pages 237 - 42 , Vancover, June 1989 .

Mark Johnson. Attribute-Value Logic and the Theory of Grammar. CSLI Lecture Notes no. 16. Center fro the Study of Language and Information, Stanford, CA, 1988 .

Mark Johnson. A left-corner program transformation for natural language parsing. (forthcoming) .

R. Kaplan and J. Bresnan . Lexical functional grammar : a formal system for grammatical representation . In Bresnan , ed. The Mental Representation of Grammatical Relations , pages 173-281. MIT Presss, Cambridge, MA, 1982 .

Y. Matsumoto, H. Tanaka, H. Hirikawa, H. Miyoshi, and H. Yasukawa. BUP: A bottom-up parser embedded in Prolog. New Generation Computing 1(2):145-158, 1983 .

Fernando C. N. Pereira and Stuart M. Shieber. Prolog and Natural Language Analysis. CSLI Lecture Note No. 10. CSLI, Stanford , CA , 1987 .

Fernando, C . N. Pereira and David H. D. Warren . Parsing as deduction . In Proceedings of the 21th Annual Meeting of the Association for Computational Linguistics, pages 137-144, MIT Cambridge, MA, June 1983 .

D. J. Rosencrantz and P. M. Lewis . Deterministic left-corner parsing . In Eleventh Annual Symposium on Switching and Automata Theory , pages 139 - 153. IEEE , 1970 . Extended Abstract .

Stuart M. Shieber . Constraint-Based Grammar Formalisms . MIT Press , Cambridge, MA, 1992 .

Figure 5.40: Unseen Document 9605023, Automatic Argumentative Zoning by Naive Bayes (2 of 2)

A Simple Transformation for Offline-Parsable Grammars and its Termination Properties

Marc Dymetman -- 9605023 -- Coling 94

Abstract

A-0 We present, in easily reproducible terms , a simple transformation for offline-parsable grammars which results in a provably terminating parsing program directly top-down interpretable in Prolog . A-1 The transformation consists in two steps : A-2 removal of empty productions , followed by : A-3 left-recursion elimination . A-4 It is related both to left-corner parsing (where the grammar is compiled , rather than interpreted through a parsing program , and with the advantage of guaranteed termination in the presence of empty productions) and to the Generalized Greibach Normal Form for DCGs (with the advantage of implementation simplicity) .

Motivation

S-0 Definite clause grammars (DCGs) are one of the simplest and most widely used unification grammar formalisms. S-1 They represent a direct augmentation of context-free grammars through the use of (term) unification (a fact that tends to be masked by their usual presentation based on the programming language Prolog) . S-2 It is obviously important to ask whether certain usual methods and algorithms pertaining to CFGs can be adapted to DCGs , and this general question informs much of the work concerning DCGs , as well as more complex unification grammar formulisms (to cite only a few areas : Earley parsing , LR parsing , left-corner parsing , Greibach Normal Form) .

S-3 One essential complication when trying to generalize CFG methods to the DCG domain lies in the fact that , whereas the parsing problem for CFGs is decidable , the corresponding problem for DCGs is in general undecidable . S-4 This can be shown easily as a consequence of the noteworthy fact that any definite clause program can be viewed as a definite clause grammar " on the empty string " , that is , as a DCG where no terminals other than <EQN/> are allowed on the right-hand side of rules . S-5 The Turing - completeness of definite clause programs therefore implies the undecidability of the parsing problem for this subclass of DCGs , and a fortiori for DCGs in general . S-6 In order to guarantee good computational properties for DCGs , it is then necessary to impose certain restrictions on their form such as offline - parsability (OP) , a nomenclature introduced by Pereira and Warren 1983 , who define an OP DCG as a grammar whose context-free skeleton CFG is not infinitely ambiguous , and show that OP DCGs lead to decidable parsing problem .

S-7 Our aim in this paper is to propose a simple transformation for an arbitrary OP DCG putting it into a form which leads to the completeness of the direct top-down interpretation by the standard Prolog interpreter : parsing is guaranteed to enumerate all solutions to the parsing problem and terminate . S-8 The existence of such a transformation is known : in Dymetman 1992a , Dymetman 1992b , we have recently introduced a " Generalized Greibach Normal Form " (GGNF) for DCGs, which leads to termination of top-down interpretation in the OP case. S-9 However , the available presentation of the GGNF transformation is rather complex (it involves an algebraic study of the fixpoints of certain equational systems representing grammars .) . S-10 Our aim here is to present a related , but much simpler , transformation , which from a theoretical viewpoint performs somewhat less than the GGNF transformation (it involves some encoding of the initial DCG , which the GGNF does not , and it only handles offline-parsable grammar , while the GGNF is defined for arbitrary DCGs) , but in practice is extremely easy to implement and displays a comparable behaviour when parsing with an OP grammar .

S-11 The transformation consists of two steps : S-12 empty-production elimination and S-13 left-recursion elimination .

S-14 The empty-production elimination algorithm is inspired by the usual procedure for context-free grammars . S-15 But there are some notable differences , due to the fact that removal of empty-productions is in general impossible for non-OP DCGs. S-16 The empty-production elimination algorithm is guaranteed to terminate only in the OP case. S-17 It produces a DCG declaratively equivalent to the original grammar .

S-18 The left-recursion elimination algorithm is adapted from a tranformation proposed in Dymetman et al. 1990 in the context of a certain formalism (" Lexical Grammars ") which we presented as a possible basis for building reversible grammars . S-19 The key observation (in slightly different terms) was that , in a DCG , if a nonterminal g is defined literally by the two rules (the first of which is left-recursive) :

[IMAGE]

S-20 then the replacement of these two rules by the three rules (where <EQN/> is a new nonterminal symbol , which represents a kind of " transitive closure " of d) :

[IMAGE]

S-21 presents the declarative semantics of the grammar .

S-22 We remarked in Dymetman et al. 1990 that this transformation is closely related to left-corner parsing " , but did not give details . S-23 In a recent paper Johnson forthcoming introduces " a left-corner program transformation for natural language parsing " , which has some similarity to the above transformations , but which is applied to definite clause grammars, rather than DCGs . S-24 He proves that this transformation respects declarative equivalence , and also shows , using a model -theoretic approach , the close connection of his transformation with left-corner parsing Rosenkrantz and Lewis 1970 , Matsumoto et al. 1983 , Pereira and Shieber 1987 .

S-25 It must be noted that the left-recursion elimination procedure can be applied to any DCG ,whether OP or not. S-26 Even in the case where the grammar is OP , however , it will not lead to a terminating parsing algorithm unless empty productions have been prealably eliminated from the grammar , a problem which is shared by the usual left-corner parser-interpreter .

S-27 Due to the space available , we do not give here correctness proofs for the algorithm presented , but expect to publish them in a fuller version of this paper . S-28 These algorithms have actually been implemented in a slightly extended version , where they are also used to decide whether the grammar proposed for transformation is in fact offline-parsable or not .

Figure 5.41: Unseen Document 9605023, Automatic Argumentative Zoning by Bigram (1 of 2)

Empty-production elimination

S-29 It can be proven that , if DCG0 is an OP DCG , the following transformation , which involves repeated partial evaluation of rules that rewrite into the empty string , terminates after a finite number of steps and produces a grammar DCG without empty-productions which is equivalent to the initial grammar on non-empty strings :

[IMAGE]

S-30 For instance the grammar consisting in the nine rules appearing above the separation in fig. <CREF/> is transformed into the grammar (see figure) :

[IMAGE]

Left-recursion elimination

S-31 The transformation can be logically divided into two steps: S-32 an encoding of DCG into a "generic " form DCG' , and S-33 a simple replacement of a certain group of left-recursive rules in DCG' by a certain equivalent non left-recursive group of rules , yielding a top-down interpretable DCG'' . S-34 An example of the transformation <EQN/> is given in fig. <CREF/> .

S-35 The encoding is performed by the following algorithm :

{IMAGE}

S-36 The procedure is very simple . S-37 It involves the creation of a generic nonterminal g(X) , of arity one , which performs a task equivalent to the original nonterminals <EQN/> . S-38 The goal <EQN/> , for instance , plays the same role for parsing a sentence as did the goal <EQN/> in the original grammar .

S-39 Two further generic nonterminals are introduced : t(X) accounts for rules whose right-hand side begins with a terminal , while d(Y,X) accounts for rules whose right-hand side begins with a non-terminal. S-40 The rationale behind the encoding is best understood from the following examples , where <EQN/> represents rule rewriting :

[IMAGE]

S-41 The second example illustrates the role played by d(Y, X) in the encoding. S-42 This nonterminal has the following interpretation : X is an " immediate " extension of Y using the given rule . S-43 In other words , Y corresponds to an " immediate left corner " of X .

S-44 The left-recursion elimination is now performed by the following " algorithm " :

[IMAGE]

S-45 In this transformation , the new nonterminal <EQN/> plays the role of a kind of transitive closure of d . S-46 It can be seen that , relative to DCG'' , for any string w and for any ground term z , the fact that g(z) rewrites into w -- or , equivalently , that there exists a ground term x such that <EQN/> rewrites into w -- is equivalent to the existence of a sequence of ground terms <EQN/> and a sequence of strings <EQN./> such that t(x1) rewrites to w1, d(x1, x2) rewrites into w2, ... , d(xk -1, xk) rewrites into wk , and such that w is the string concatenation <EQN/> . S-47 From our previous remark on the meaning of d(Y, X) , this can be interpreted as saying that " constituent x is a left-corner of constituent z " , relatively to string w .

S-48 The grammar DCG'' can now be compiled in the standard way -- via the adjunction of two " differential list " arguments -- into a Prolog program which can be executed directly. S-49 If we started from an offline-parsable grammar DCG0 , this program will enumerate all solutions to the parsing problem and terminate after a finite number of steps .

References

Marc Dymetman . A Generalized Greibach Normal Form for Definite Clause Grammars . In proceedings of the 15th International Conference on Computational Linguistics, volume 1, pages 366-372, Nantes, France, July 1992 .

Marc Dymetman. Transformations de grammaires logiques et reversibilites en Traduction Automatique . These d'Etat , 1992 . Universite Joseph Fourier (Grenoble 1) , Grenoble , France .

Marc Dymetman and Pierre Isabelle . Reversible logic grammars for machine translation . In Proceedings of the Second International Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages, Pittsburgh, PA, June 1988. Carnegie Mellon University.

Marc Dymetman, Pierre Isabelle, and Francois Perrault. A symmetrical approach to parsing and generation. In Proceedings of the 13th International Conference on Computational Linguistics, volume 3, pages 90-96, Helsinki , August 1990 .

Andrew Haas. A generalization of the offline-parsable grammars. In Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics, pages 237 - 42 , Vancover, June 1989 .

Mark Johnson. Attribute-Value Logic and the Theory of Grammar. CSLI Lecture Notes no. 16. Center fro the Study of Language and Information, Stanford, CA, 1988 .

Mark Johnson. A left-corner program transformation for natural language parsing. (forthcoming) .

R. Kaplan and J. Bresnan . Lexical functional grammar : a formal system for grammatical representation . In Bresnan , ed. The Mental Representation of Grammatical Relations , pages 173-281. MIT Presss, Cambridge, MA, 1982 .

Y. Matsumoto, H. Tanaka, H. Hirikawa, H. Miyoshi, and H. Yasukawa. BUP: A bottom-up parser embedded in Prolog. New Generation Computing 1(2):145-158, 1983 .

Fernando C. N. Pereira and Stuart M. Shieber. Prolog and Natural Language Analysis. CSLI Lecture Note No. 10. CSLI, Stanford , CA , 1987 .

Fernando, C . N. Pereira and David H. D. Warren . Parsing as deduction . In Proceedings of the 21th Annual Meeting of the Association for Computational Linguistics, pages 137-144, MIT Cambridge, MA, June 1983 .

D. J. Rosencrantz and P. M. Lewis . Deterministic left-corner parsing . In Eleventh Annual Symposium on Switching and Automata Theory , pages 139 - 153. IEEE , 1970 . Extended Abstract .

Stuart M. Shieber . Constraint-Based Grammar Formalisms . MIT Press , Cambridge, MA, 1992 .

Figure 5.42: Unseen Document 9605023, Automatic Argumentative Zoning by Bigram (2 of 2)

	Seen	Unseen
Cue Phrase Feature	60.9	54.9
All Features	71.6	65.3
Baseline	29.1	

Figure 5.43: Performance of Meta-Discourse Features; Unseen and Seen Data

were compiled from the other two parts. The advantage of this was that we now had gold standards for the "unseen" part, and we could compare the system's performance with both lists. Performance decreased significantly on unseen data, but not catastrophically, as can be seen from figure 5.43 (values refer to relevance-extraction, and are given in precision = recall values, in Kupiec et al. style). Even though the task is not the same, and the cue phrase method has been improved since to form our more recent meta-discourse features `Formu`, `Ag-1` and `Ag-2`, we still conclude from this experiment that meta-discourse features can be rather stable, even if only two thirds of the data is taken into account.

5.6. Conclusion

Annotator	Kappa	Raw Agr.	Random Agr.
System:			
Naive Bayes	.39	71%	54%
Naive Bayes + Bigram	.41	70%	49%
Humans:			
Task-trained	.71	87%	56%
Non task-trained (avg.)	.51	76%	49%
Baselines:			
Most frequent category	-.12	68%	71%
Random, uniform distribution	-.10	14%	22%
Random, observed distribution	0	48%	48%

Figure 5.44: Results of Human and Automatic Argumentative Zoning, I

Figures 5.44 and 5.45 summarize all evaluation results. If we compare humans and automatic results we see that there is still plenty of room for improvement for our systems. However, the automatic performance results are also a lot better than random, as the distance from the K=0 point (the most sensible baseline for our task) shows. Argumentative Zoning is a new task, so there are no direct numerical values to compare our prototype's performance with. When compared to Kupiec et al.'s result, both an earlier implementation (Teufel and Moens, 1997) and the current results compare favourably, if we consider our systems' success on AIM sentences. Additionally, if all one wants are extracted AIM and TEXTUAL sentences, our symbolic rules provide a good solution: both our implementations are much better at categorizing TEXTUAL and AIM sentences than they are at categorizing BASIS and CONTRAST sentences.

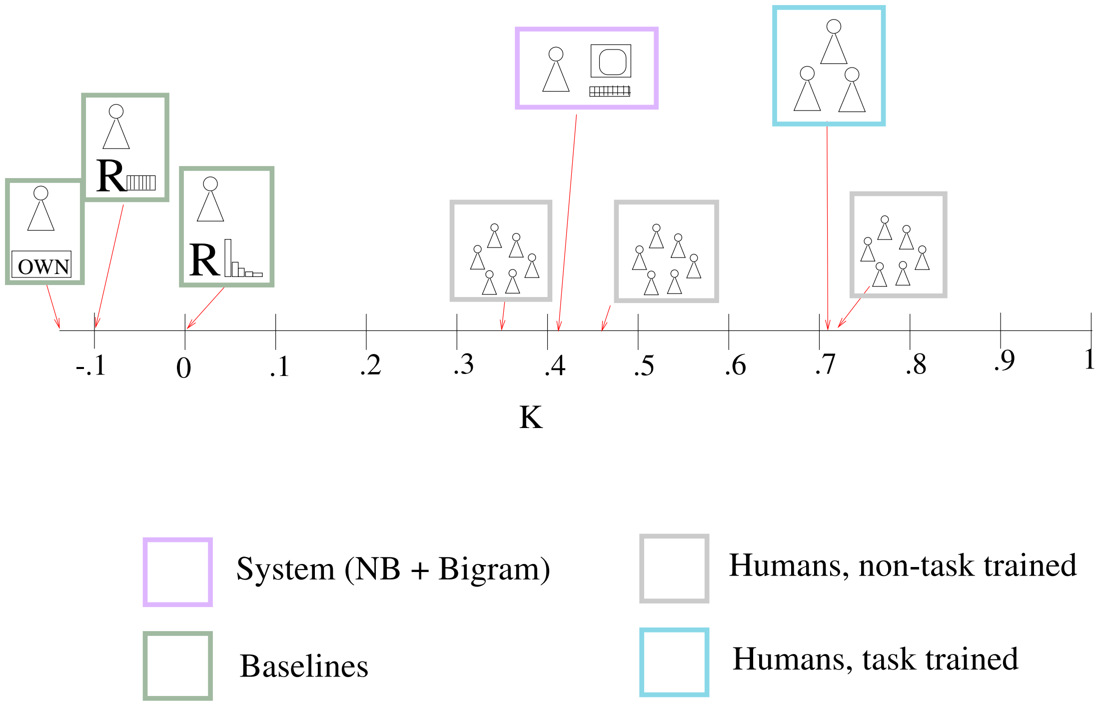

Figure 5.45: Results of Human and Automatic Argumentative Zoning, II

However, statistical classification is still rather noisy. We assume that the main reason for this is lack of training data: we were training on only 72 documents. However, as corpus collection and manual annotation with such a high level of document semantics is rather time consuming, it was not possible in the time frame of this thesis to expand the training data.

We believe that numerically high results are not absolutely required for a workable system. We see Argumentative Zoning as a forgiving task. Language is redundant, and the most important pieces of information will be repeated in the paper. Names of other peoples' solutions, for example, or references to based-on solutions, get repeated over and over—recognizing them *once* is enough to get the right kind of information into our RDP slot. We often found in the human annotation experiment that different versions of annotation on one paper still essentially contained the same information, i.e. would have resulted in similar RDPs. This effect would probably also apply to papers which are less than optimally zoned by an automatic process.

We see our results as an indication that we are on the right track for a difficult task, even though they are still modest at present. Some of the features known from text extraction have reconfirmed their usefulness for a new task. Our new features for argumentative sentence classification, which are based on agents and actions, have managed to increase our statistical results, and they have also provided useful input to the symbolic classification results.

Chapter 6

Conclusions

In this thesis, we have introduced a new task for document management, which we call *Argumentative Zoning*. Argumentative Zoning is the analysis of the argumentative status of sentences in scientific articles. Figure 6.1 shows how argumentative zones (and their derivatives, RDPs or Rhetorical Document Profiles) act as intermediaries between the reader and the writer. It also shows the setup of the experiments we performed to explore the task of Argumentative Zoning: a system for automatic Argumentative Zoning is evaluated intrinsically by comparison to human Argumentative Zoning. At the same time, the human annotation provides training material for the system.

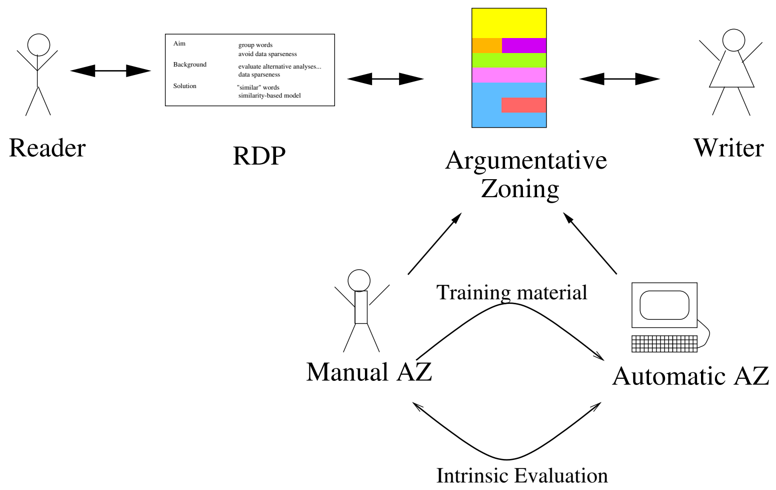

Figure 6.1: Overview of Argumentative Zoning Experiments

6.1. Contribution of the Thesis

The main theoretical claim of this thesis is that empirical discourse analysis can contribute towards the problem of document characterization in a document retrieval environment. We exemplify this by applying an analysis of prototypical scientific argumentation, Argumentative Zoning, to scientific articles. We claim that the type of document structure that argumentative zones capture is dominant in this text type, and also particularly useful for our task.

While Argumentative Zoning relies on rhetorical effects which are specific to the text type, it is independent of the subject matter treated. We have shown that the task of Argumentative Zoning is defined well enough for humans to be able to perform it consistently.

We have identified sentential features which correlate with the argumentative status of the given sentence. The existence of these correlates means that human annotation behaviour can in principle be simulated automatically. We have provided algorithms for the determination of these features. The more complicated features aim at modelling meta-discourse as an expression of prototypical scientific argumentation; we use linguistic heuristics and pattern matching to this end.

The practical contributions of this thesis are threefold:

- *Corpus collection* (section 5.3.2): we have collected and XML-encoded a substantial amount of unrestricted, "naturally occurring" scientific text from a scientific web archive. As collection proceeded in an unbiased way, we expect the corpus to be representative for the source.

- *Development of annotation scheme for Argumentative Zoning* (section 3.3): we have defined an annotation scheme for the argumentative status of sentences which is consistent and informative. The reproducibility and stability of the annotation scheme was evaluated by an experiment with two unrelated, task trained human annotators (section 4.3).

- *Implementation of a prototype system for automatic Argumentative Zoning*: we have provided evidence that this annotation scheme can be automatically applied (chapter 5). The prototype uses supervised learning on the basis of the previously hand-annotated corpus. The approach relies on corpus-based robust features, well-known from traditional text extraction work, but it is accompanied by a new, more linguistically motivated pattern matching to find prototypical agents and actions.

We have argued in chapter 2 that RDPs (Rhetorical Document Profiles) are document profiles which are specially useful for partially informed readers in a DR environment, and that they can be used for the production of tailored summaries and more informative citation information. Argumentative Zoning, as explored in this thesis, is a necessary and useful subtask for the generation of RDPs; however, this thesis does not accomplish the generation of RDPs. In the next section, we will sketch which tasks still need to be done in order to construct RDPs.

6.2. Future Work

6.2.1. RDP Generation

One avenue of future work is obvious: the algorithm for actually creating RDPs is not implemented yet. However, we have already given the outline of the two main parts of the algorithm:

- Determination of most appropriate slot fillers (in section 2.1.1);
- Association of identifiers of other approaches with the sentence expressing author's stance (in section 3.4). More advanced approaches for this subtask are discussed in the following.

Similarity matching between sentences could be used to determine the best filler for those slots which are filled by entire sentences (e.g. BACKGROUND). Different similarity measures are imaginable, from simple surface based algorithms like the Longest Common Substring as used by us in earlier work (cf. section 4.1.2.2), to more complicated ones like LIKEIT (Yianilos, 1997). Similarity as defined by vector space models is another option (Salton, 1971). One could, however, apply a deeper approach based on agent and action comparison, similar to Barzilay et al.'s (1999) work, and we would advocate this.

Given the stage of development reached in the thesis, extrinsic evaluation would be premature. Eventually, we envisage a task-based evaluation scenario, where the performance of subjects using RDPs for a certain task (e.g. question answering or relevance decision) is compared to a control group working with sentence extracts, and a group working with full documents. Such evaluation needs a clear definition of the task of information foraging for uninformed readers. The right task definition is not easy to find, particularly as user studies concentrating on this user group are rare (chapter 2). We are convinced at this point that simple relevance decision is under-defined and cannot be used as a task; we expect that a clearer picture of the best task for extrinsic evaluation will emerge during the actual generation of RDPs.

6.2.2. Improving the Prototype

We have shown in chapter 5 that it is possible to find patterns in the extracted sentential features with a relatively simple implementation and simple statistical techniques. As a result, our system can simulate human annotation behaviour to a certain degree. However, there are many aspects in which the existing prototype could be improved.

One could imagine a cascading system which performs an analysis of the agent-and-action structure of the text prior to the classification of the full annotation scheme. The first step, the attribution of intellectual ownership, could be learned from text annotated with the basic annotation scheme, by associating the patterns with agents (US_AGENT—THEM_AGENT—GENERAL_AGENT). In a second step, the finer distinctions could be applied.

In a cascading system, the high-precision rules described in section 5.3.5 could act as "sure-fire" rules: evidence of different levels of certainty could be collected before a statistically-based search, and "sure-fire" rules could provide the starting point, similar to the system presented by Mikheev et al. (1998).

In particular the actions are a topic which requires more research. We have created the action lexicon (figure 5.8; page 195) manually, based only on our intuitions after inspecting the corpus. But no clear methodology for creating the lexicon has emerged yet. We would like to perform tests varying the verbs included in the action lexicon and the classes assigned. Independent information sources like Levin's (1993) alternation classes, or WordNet (Klavans and Kan, 1998) could be used. And a more systematic way to create this lexicon would be to use learning in a bottom-up way.

We observed problems with verbal ambiguity: the same verbs are sometimes used in a meta-discourse interpretation and sometimes not. This is illustrated by the following examples:

CONTINUATION_ACTION:
For our analysis of gapping, we follow Sag (1976) in hypothesizing [...] (S-38, 9405010)

Not a CONTINUATION_ACTION:
From this or-node we follow an arc labelled Id [...] (S-73, 9405022)

CONTRAST_ACTION:
Hobbs' ordering of entities from a previous utterance varies from Brennan et al.'s [...] (S-104, 9410006)

Not a CONTRAST_ACTION:
The number of test contexts varies from word to word [...] (S-78, 9503025)

The examples seem to imply that an analysis of the syntactic context, in this case, the direct object, might help, but we fear the problem lies deeper. Given that we want to avoid the need for full text comprehension, traditional Word Sense Disambiguation (Schütze, 1998; Yarowsky, 1995) might help.

Apart from verbal polysemy, there are some other specific concepts which supposedly indicate meta-discourse, but which are problematic for our approach, e.g. "*goals*", "*topic*" and "*similarity*". These concepts are used at the object level (*science*) in some papers, e.g. in logic programming, discourse modelling and in statistical NLP:

The speaker attempts to achieve this goal by building a description of the object that she believes will give the hearer the ability to identify it when it is possible to do so. (S-6, 9405013)

The substructure check makes only sense if the semantics <EQN/> of the current goal is instantiated. (S-69, 9405004)

The sentential topic Hanako is the only possible antecedent of this zero subject in this example. (S-13S, 9405028)

In those models, the relationship between given words is modeled by analogy with other words that are in some sense similar to the given ones. (S-11, 9405001)

In experiments not reported here in detail, we have tried to ameliorate this problem by excluding those `Ag-1`, `Ag-2` and `Formu` patterns which contain "characteristic" words for this document, as determined by a *tf/idf* measure. The idea was that if a phrase which we intended to indicate meta-discourse occurred far more often than expected in a given document, then there was a chance that it is a concept at an object level. However, these experiments did not result in higher recognition results. We have to conclude that this is another problem which requires further enquiry.

Finding identifiers of other work is important for building RDPs (cf. above). Whereas this task is easy in the cases where a formal citation is present, it is much harder to identify well-known names of solutions in text, e.g. as in the following sentence:

> *I argue that Hidden Markov Models are unsuited to the task* [...]
> (S-9, 941002)

Only later in the text, *"Hidden Markov Models"* are associated with particular researchers:

> *Hidden Markov Models (HMMs) (Huang et al., 1990) offer a powerful statistical approach to this problem* [...] (S-24, 941002)

However, the identification of "*Hidden Markov Models*" as a solution name would have several advantages in this context:

- The names would be fillers of the RDP slots "SOLUTION ID" (parts of the complex slots BASIS/CONTINUATION and RIVAL/CONTRAST). Such a characterization of other work is more informative than formal citations in many cases, as names of solutions have more continuity than single papers and single researchers.

- A list of such names could help the uninformed reader acquire an overview of the field (cf. chapter 1). Names of commonly advocated solutions might help identify schools of thought, in this case, groups of researchers who have invented Hidden Markov Models or who work with them. Named problems, e.g. *"data sparseness"* also occur frequently in our texts, and their identification would be similarly useful to uninformed readers.

- Identifying names of solutions would help improve the agent feature, as researchers' names are often substituted with (named) approaches or solutions they are well-known for. At the moment, the sentence above would not be classified as part of prototypical argumentation, because the agent is not recognized as THEM_AGENT, but if the authors had used the expression "*Huang et al.'s (1990) approach*" it would. This lack of parallelism makes the method less robust towards writing style.

Recent advances in named entity recognition have made the association task technically feasible, cf. the results of the Named Entity Recognition Task in MUC-7, where F-measures are in the range of 93% for domain-specific text (MUC-7, 1998).

Note that there are typically contexts in the article where the association of "THEM" or "US" with a solution name is easier than in other contexts. Consider the following sentence:

> *LHIP provides a processing method which allows selected portions of the input to be ignored or handled differently.* (S-5, 9408006)

This sentence (and the role of *"LHIP"* in the argumentation) can only be understood in the context of a sentence several sentences earlier:

> *This paper describes LHIP (Left-Head Corner Island Parser), a parser designed for broad-coverage handling of unrestricted text.* (S-0, 9408006)

The sentence would have to be interpreted completely differently in the context of the following (imaginary) sentence:

> *Gold et al. (1989) introduced LHIP (Left-Head Corner Island Parser), a parser designed for broad-coverage handling of unrestricted text.*

Recognition of *"LHIP"* in close proximity with the phrase *"in this paper"* could add *"LHIP"* to a list of solutions associated with the authors, whereas in the other (fictional) case, it would have been added to a list of approaches associated with Gold et al. (THEM_AGENT).

There is one other possibility how agent recognition could be made more robust, and that is by anaphora resolution. As reported in section 5.2.2.2, not all agent classes are ambiguous. In fact, in many of them, interpretation is unambiguous (THEM_AGENT, US_AGENT); in others, we have found a strong tendency that the intended interpretation is almost always present (TEXTSTRUCTURE_AGENT, OUR_AIM_AGENT, US_PREVIOUS_AGENT, REF_US_AGENT, GAP_AGENT, SOLUTION_AGENT, PROBLEM_AGENT). However, a high level of ambiguity is associated with the classes REF_US_AGENT, THEM_PRONOUN_AGENT, AIM_REF_AGENT, REF_AGENT. Most of these ambiguities are between US_AGENT and THEM_AGENT, but the agent class THEM_PRONOUN_AGENT is actually ambiguous between THEM_AGENT and any plural objects in the scientific domain the paper is talking about, e.g. rules, arcs, probabilities. Examples for correct and incorrect interpretation of THEM_PRONOUN_AGENTs can be found in appendix B.7; p. 300. For example, agents no. 4 and 16 have the wrong interpretation.

We performed a simulation experiment to determine the distribution of US_AGENT, THEM_AGENT and GENERAL_AGENT for the most frequent of the ambiguous classes, REF_AGENT. There were 632 occurrences of REF_AGENT in the corpus (only 586 of which were used in the Naive Bayesian classification and the symbolic rules; the others were not the first agent in the sentence). We wanted to determine if anaphora resolution prior to classification would improve end results, so we manually simulated a perfect anaphora resolution algorithm by classifying the phrases by their referent: 436 (69%) of the 632 REF_AGENTs were classified as US_AGENT, 175 (28%) as THEM_AGENT, and 20 (3%) as GENERAL_AGENT.

As a result of this manual disambiguation, the performance of the `Ag-1` feature for the Naive Bayesian model increased dramatically from K=.07 to K=.14, making it the third best feature after `Cit-1` (K=.18) and `Loc` (K=.17); cf. figure 5.33 (p. 222). Classification results using the 14 successful features increased from K=.39 to K=.42. These results are surprisingly good, considering that we removed only one ambiguous class. Even though a practical anaphora resolution model would not achieve 100% correctness as we did in our simulation, our experiment still points to the fact that good anaphora resolution would make statistical classification less noisy by potentially removing the need for ambiguity classes, and that it could potentially be of great value for automatic Argumentative Zoning.

6.2.3. Learning Meta-discourse Expressions

The current experiments have shown that sentential features, particularly meta-discourse phrases, can help us perform Argumentative Zoning. It is a practical problem of how to arrive at good patterns other than manually generating them. There are some approaches which learn cue phrases automatically from text, either by ngram-techniques (Samuel et al., 1998, 1999) or by *tf/idf* style frequency techniques (Hovy and Lin, 1999; Hovy and Liu, 1998). Learning would be particularly useful for the clustering of values, which we have so far done manually. We performed some experiments with n-grams over words as approximations for indicator phrases (Teufel, 1998); these experiments showed over-fit and were thus not conclusive.

We take this as an indication that our corpus is still too small to automatically learn good patterns. The learning of agent and action patterns, however, is planned for the future, when our corpus of scientific articles will hopefully be expanded considerably.

6.2.4. Redefining the Annotation Task

The task of Argumentative Zoning could be refined by using a more fine-grained unit of annotation and classification. Currently, we use *sentences*; part of the reason for this decision was practical, as sentence boundary disambiguators like the one we use work very reliably. However, we came across many examples where a border between two argumentative zones cuts across a sentence:

> *However, this is not very satisfactory because one of the goals of our work is precisely to avoid the problems of data sparseness by grouping words into classes.* (S-41, 9408011)
>
> *While we know of previous work which associates scores with feature structures (Kim, 1994)* [sic] *are not aware of any previous treatment which makes explicit the link to classical probability theory.* (S-9, 9502022)

In the first case, there is a borderline between a CONTRAST and an AIM zone which cuts across the sentence, in the second between an OTHER and CONTRAST zone. Cases like this confuse both symbolic and stochastic accounts of Argumentative Zoning, as correlates of both zones can be found in the sentence, but only one target outcome is annotated.

Our experience with the heuristics for action and agent detection in sections 5.3.3.7 have shown that it is theoretically possible to dissect the sentence into clause-like units—though we have so far used this information only for feature determination. These heuristics rely only on the most likely finite verbs in the sentence as determined by a POS-Tagger. Even though a definition of a clause as centered around a finite verb is simplistic (cf. also the discussions in section 3.5 in the context of RST), and even though such heuristics are not correct in all cases, we nevertheless argue that a clause-based approach would have advantages for Argumentative Zoning. The finer unit of annotation is intuitively more appealing, as clauses map more directly to propositions. A move towards the clause would thus be a move towards a slightly deeper representation.

Another way to improve the task of Argumentative Zoning would be to ask the subjects to indicate a relevance-level (or confidence-level) for the annotation of each sentence. This would indicate how well suited the sentence is to serve as an RDP slot. Of course, such instructions would result in a higher training effort, but would also provide us with a more valuable gold standard for the task.

6.2.5. Application to a Different Domain

Finally, we take a look at the kinds of texts treated. We have assumed that argumentative moves and zones are to be expected in *all* scientific research articles, as they are based on the function associated with the text type, i.e. the goal of justifying the validity of the research presented. We have concluded from this that our annotation scheme should in principle apply to all kinds of scientific research articles. One of the reasons for choosing computational linguistics articles was the interdisciplinary nature of the field, which would make the corpus a difficult test bed. Nevertheless, our claim would find a more rigorous verification if we could successfully apply the analysis to texts of a different domain.

It is plausible that some of the meta-discourse we found is specific to our corpus. Research by Hyland (1998) confirms that there are differences in meta-discourse between domains. In that case, an approach which learns new cue phrases from text, as mentioned above, would be particularly useful for porting our implementation to a new domain.

It might also be the case that our young, interdisciplinary domain contains particularly many argumentative moves of explicit comparison. In such domains, contrast with other researchers and intellectual ancestry is very important, as there are many methodologies, which are often identified by similarities to and contrast with existing ones. It might thus be the case that other domains do not express comparisons to other work as overtly as our texts do.

We have used *conference* articles in this thesis. Practical reasons have kept us from using journal articles as data so far: the difficulty of corpus collection due to copy right problems, and due to the increased length and subsequent time effort of human experiments. In principle, however, we are particularly interested in journal articles, for several reasons. On the one hand, they can be expected to be of higher textual quality, as they are more rigorously edited. On the other hand, as journal articles are much longer, they pose a particularly difficult problem for current summarization approaches, as these do not take large-scale discourse structure into account. As the scientific argumentation in journal articles is basically the same as in conference articles, we are confident that our scheme should be applicable to journal articles at least as consistently as to conference articles.

Bibliography

Abney, Steven. 1990. Rapid incremental parsing with repair. In *Proceedings of the 6th New OED Conference*, 1–9.

Abracos, Jose, and Gabriel Pereira Lopes. 1997. Statistical methods for retrieving most significant paragraphs in newspaper articles. In Mani and Maybury 1997, 51–57.

ACP online. 1997. Annals Extracts. `http://www.acponline.org/journals/annals/01apr97/extracts/extractintro.%htm`.

Adhoc. 1987. Ad Hoc Working Group For Critical Appraisal Of The Medical Literature. A proposal for more informative abstracts of clinical articles. *Annals of Internal Medicine* 106: 508–604.

Adler, Annette, Anuj Gujar, Beverly L. Harrison, Kenton O'Hara, and Abigail Sellen. 1998. A diary study of work-related reading: Design implications for digital reading devices. In *Proceedings of CHI-98, ACM*, 241–248.

Alexandersson, Jan, Elisabeth Maier, and Norbert Reithinger. 1995. A robust and efficient three-layered dialogue component for a speech-to-speech translation system. In *Proceedings of the Seventh Meeting of the European Chapter of the Association for Computational Linguistics*, 188–193.

Alley, Michael. 1996. *The Craft of Scientific Writing*. Englewood Cliffs, NJ: Prentice-Hall.

Alterman, Richard. 1985. A dictionary based on concept coherence. *Artificial Intelligence* 25(2): 153–186.

ANSI. 1979. American National Standard for Writing Abstracts. Technical report, American National Standards Institute, Inc., New York, NY. ANSI Z39.14.1979.

Arndt, Kenneth A. 1992. The informative abstract. *Archives of Dermatology* 128(1): 101.

Baldwin, Breck, and Tom Morton. 1998. Dynamic coreference-based summarization. In *Proceedings of the Third Conference on Empirical Methods in Natural Language Processing (EMNLP-3)*.

Baldwin, Breck, Tom Morton, Amit Bagga, Jason Baldridge, Raman Chandrasekar, Alexis Dimitriadis, Kieran Snyder, and Magdalena Wolska. 1998. Description of the UPenn CAMP system as used for coreference. In MUC-7 1998.

Barzilay, Regina, and Michael Elhadad. 1999. Using lexical chains for text summarization. In Mani and Maybury 1999, 111–121.

Barzilay, Regina, Kathleen R. McKeown, and Michael Elhadad. 1999. Information fusion in the context of multi-document summarization. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL-99)*, 550–557.

Bates, Marcia J. 1998. Indexing and access for digital libraries and the internet: Human, database and domain factors. *Journal of the American Society for Information Science* 49: 1185–1205.

Baxendale, Phyllis B. 1958. Man-made index for technical literature—an experiment. *IBM Journal of Research and Development* 2(4): 354–361.

Bazerman, Charles. 1985. Physicists reading physics, schema-laden purposes and purpose-laden schema. *Written Communication* 2(1): 3–23.

Bazerman, Charles. 1988. *Shaping Writing Knowledge*. Madison, WI: University of Wisconsin Press.

Belkin, N. 1980. Anomalous states of knowledge as a basis for information retrieval. *The Canadian Journal of Information Science* 5: 133–143.

Biber, Douglas, and E. Finegan. 1994. Intra-textual variation within medical research articles. In Oostdijk and de Haan, eds., *Corpus-Based Research into Language*, chapter 13, 201–221. Amsterdam: Rodoph.

Blicq, Ron. 1983. *Technically-write!: Communicating in a Technological Era*. Scarborough, Ont.: Prentice-Hall Canada.

Boguraev, Branimir, and Christopher Kennedy. 1999. Salience-based content characterization of text documents. In Mani and Maybury 1999, 99–110.

Bonzi, Susan. 1982. Characteristics of a literature as predictors of relatedness between cited and citing works. *Journal of the American Society for Information Science* 33(4): 208–216.

Borgman, Christine L. 1996. Why are online catalogs still hard to use? *Journal of the American Society for Information Science* 47: 493–503.

Borko, Harold, and C. L. Bernier. 1975. *Abstracting Concepts and Methods*. San Diego, CA: Academic Press.

Borko, Harold, and Seymour Chatman. 1963. Criteria for acceptable abstracts: A survey of abstractors' instructions. *American Documentation* 14(2): 149–160.

Brandow, Ronald, Karl Mitze, and Lisa F. Rau. 1995. Automatic condensation of electronic publications by sentence selection. *Information Processing and Management* 31(5): 675–685.

British Telecom. 1998. `http://transend.labs.bt.com/cgi-bin/prosum/prosum`.

Broer, J. W. 1971. Abstracts in block diagram form. *IEEE Transactions on Engineering Writing and Speech* 14(2): 64–67.

Brooks, Terrence A. 1986. Evidence of complex citer motivations. *Journal of the American Society for Information Science* 37: 34–36.

Brouwer, M., C. C. Clark, G. Gerbner, and K. Krippendorff. 1969. The television world of violence. In *Mass Media and Violence: A Report to the National Commission on the Causes and Prevention of Violence*, 311–339 and 519–591. Washington, D.C.: Government Printing Office. Cited after Krippendorff:80.

Brown, Ann L., and Jeanne D. Day. 1983. Macrorules for summarizing text: The developments of expertise. *Journal of Verbal Learning and Verbal Behaviour* 22: 1–14.

Brown, Penelope, and Levinson Stephen C. 1987. *Politeness: Some Universals in Language Usage*. Cambridge, England: Cambridge University Press.

Busch-Lauer, Ines A. 1995. Abstracts in German medical journals: A linguistic analysis. *Information Processing and Management* 31(5): 769–776.

Buxton, A. B., and A. J. Meadows. 1978. Categorization of the information in experimental papers and their author abstracts. *Journal of Research in Communication Studies* 1: 161–182.

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

Carletta, Jean, Amy Isard, Stephen Isard, Jacqueline C. Kowtko, Gwyneth Doherty-Sneddon, and Anne H. Anderson. 1997. The reliability of a dialogue structure coding scheme. *Computational Linguistics* 23(1): 13–31.

Chalmers, Matthew, and Paul Chitson. 1992. Bead: Explorations in information visualization. In *Proceedings of the 15th Annual International Conference on Research and Development in Information Retrieval (SIGIR-92)*, 330–337.

Charney, Davida. 1993. A study in rhetorical reading—How evolutionists read "The Spandrels of San Marco". In Jack Selzer, ed., *Understanding Scientific Prose*. Madison, WI: The University of Wisconsin Press.

Chinchor, Nancy A., and Elaine Marsh. 1998. *MUC-7 Information Extraction Task Definition*. DARPA. `www.muc.saic.com/proceedings/muc_7_toc.html`.

Chubin, Daryl E., and S. D. Moitra. 1975. Content analysis of references: Adjunct or alternative to citation counting? *Social Studies of Science* 5(4): 423–441.

Cleverdon, Cyril W. 1984. Optimizing convenient online access to bibliographic databases. *Information Services and Use* 4: 37–47.

Clove, J. F., and B. C. Walsh. 1988. Online text retrieval via browsing. *Information Processing and Management* 24(1): 31–37.

Clyne, Michael. 1987. Cultural differences in the organization of academic texts. *Journal of Pragmatics* 11: 211–247.

CMP_LG. 1994. The Computation and Language E-Print Archive, `http://xxx.lanl.gov/cmp-lg`.

Cohen, Robin. 1987. Analyzing the structure of argumentative discourse. *Computational Linguistics* 13: 11–24.

Cohen, William W. 1995. Fast effective rule induction. In *Proceedings of the Twelfth International Conference on Machine Learning*, 115–123.

Cohen, William W. 1996. Learning trees and rules with set-valued features. In *Proceedings of AAAI-96*.

Conway, William D. 1987. *Essentials of Technical Writing*. Rexburg, Idaho: TechWrite Press. 4th ed.

Cremmins, Edward T. 1996. *The Art of Abstracting*. Arlington, VA: Information Resources Press, 2nd edn.

Crookes, Graham. 1986. Towards a validated analysis of scientific text structure. *Applied Linguistics* 7(1): 57–70.

Day, Robert A. 1995. *How to Write and Publish a Scientific Paper*. Cambridge, England: Cambridge University Press, 4th edn.

DeJong, Gerald F. 1982. An Overview of the FRUMP system. In Wendy G. Lehner and Ringle, eds., *Strategies for Natural Language Processing*, chapter 5. Hillsdale NJ: Lawrence Erlbaum.

Dillon, Andrew. 1992. Reading from paper versus from screens: A critical review of the empirical literature. *Ergonomics* 35(10): 1297–1326.

Dillon, Andrew, John Richardson, and Cliff McKnight. 1989. Human factors of journal usage and the design of electronic text. *Interacting with Computers* 1(2): 183–189.

Drakos, Nikos. 1994. From Text to Hypertext: A Post-Hoc Rationalisation of LaTeX2HTML. In *The Proceedings of the First WorldWide Web Conference*.

Dunning, Ted. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics* 19(1): 61–74.

Duszak, Anna. 1994. Academic discourse and intellectual styles. *Journal of Pragmatics* 21: 291–313.

Earl, Lois L. 1970. Experiments in automatic extracting and indexing. *Information Storage and Retrieval* 6(6): 313–334.

Edmundson, H. P. 1969. New methods in automatic extracting. *Journal of the Association for Computing Machinery* 16(2): 264–285.

Edmundson, H. P. et al. 1961. *Final Report on the Study for Automatic Abstracting*. Canoga Park, CA: Thompson Ramo Wooldridge.

Elhadad, Michael. 1993. Using Argumentation to Control Lexical Choice: A Unification-Based Implementation. Ph.D. thesis, Computer Science Department, Columbia University, New York, NY.

Ellis, D. 1989a. A behavioural approach to information system design. *Journal of Documentation* 45(3): 171–212.

Ellis, D. 1989b. A behavioural model for information system design. *Journal of*

Information Science 15(4): 237–247.

Ellis, David. 1992. The physical and cognitive paradigms in information retrieval research. *Journal of Documentation* 48: 45–64.

Endres-Niggemeyer, Brigitte, Elisabeth Maier, and Alexander Sigel. 1995. How to implement a naturalistic model of abstracting: Four core working steps of an expert abstractor. *Information Processing and Management* 31(5): 631–674.

Farr, A. D. 1985. *Science Writing for Beginners*. Oxford: Blackwell Scientific Publications.

Fidel, R. 1985. Moves in online searching. *Online Review* 9(1): 61–74.

Fidel, R. 1991. Searchers' selection of search keys. *Journal of the American Society for Information Science* 42(7): 490–527.

Finch, Steven, and Andrei Mikheev. 1995. Towards a workbench for acquisition of domain knowledge from natural language. In *Proceedings of the 7th Meeting of the European Chapter of the Association for Computational Linguistics (EACL-95)*, 194–201.

Francis, W. Nelson, and Henry Kucera. 1982. *Frequency Analysis of English Usage: Lexicon and Grammar*. Boston, MA: Houghton Mifflin.

Froom, P., and J. Froom. 1993. Deficiencies in structured medical abstracts. *Journal of Clinical Epidemiology* 46: 591–594.

Frost, Carolyn O. 1979. The use of citations in Literary Research: A preliminary Classification of Citation Functions. *Library Quarterly* 49: 405.

Garfield, Eugene. 1979. *Citation Indexing: Its Theory and Application in Science, Technology and Humanities*. New York, NY: J. Wiley.

Garfield, Eugene. 1996. The significant scientific literature appears in a amall group of journals. *The Scientist* 10(17): 13–16. See also `http://165.123.34.41/yr1996/sept/research_960902.html`.

Georgantopoulos, Byron. 1996. Automatic Summarising Based on Sentence Extraction: A Statistical Approach. Master's thesis, Dept. of Linguistics, University of Edinburgh, Edinburgh, UK.

Giles, C. Lee, Kurt D. Bollacker, and Steve Lawrence. 1998. CiteSeer: An automatic citation indexing system. In *Proceedings of the Third ACM Conference on Digital*

Libraries, 89–98.

Grefenstette, Gregory. 1994. *Explorations in Automatic Thesaurus Discovery*. Boston, MA: Kluwer Academic Press.

Grishman, Ralph, and Beth Sundheim. 1995. Design of the MUC-6 evaluation. In *Proceedings of the Sixth Message Understanding Conference*, 1–11. DARPA, San Francisco, CA: Morgan Kaufmann Publishers.

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

Grover, Claire, Andrei Mikheev, and Colin Matheson. 1999. LT TTT Version 1.0: Text Tokenisation Software. Technical report, Human Communication Research Centre, University of Edinburgh. `http://www.ltg.ed.ac.uk/software/ttt/`.

Hartley, James. 1997. Is is appropriate to use structured abstracts in social science journals? *Learned Publication* 10(4): 313–317.

Hartley, James, and Matthew Sydes. 1997. Are structured abstracts easier to read than traditional ones? *Journal of Research in Reading* 20(2): 122–136.

Hartley, James, Matthew Sydes, and Antony Blurton. 1996. Obtaining information accurately and quickly: are structured abstracts more efficient? *Journal of Information Science* 22(5): 349–356.

Haynes, R. B. 1990. More informative abstracts revisited. *Annals of Internal Medicine* 113: 69–76.

Hearst, Marti A. 1995. Tilebars: Visualization of term distribution information in full text information access. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, 59–66.

Hearst, Marti A. 1997. TextTiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics* 23(1): 33–64.

Hearst, Marti A., and Jan O. Pedersen. 1996. Reexamining the cluster hypothesis: scatter/gather on retrieval results. In *Proceedings of the 19th Annual International Conference on Research and Development in Information Retrieval (SIGIR-96)*, 76–84.

Herner, Saul. 1959. Subject slanting in scientific abstracting publications. In *Proceedings on the International Conference on Scientific Information*, vol. 1, 407–427.

Hoey, Michael. 1979. *Signalling in Discourse*. No. 6 in Discourse Analysis Monograph. Birmingham, UK: University of Birmingham.

Horsella, Maria, and Gerda Sindermann. 1992. Aspects of scientific discourse: Conditional argumentation. *English for Specific Purposes* 11: 129–139.

Houp, Kenneth W., and T. E. Pearsall. 1988. *Reporting Technical Information*. New York, NY: Maxwell Macmillan International, 6th edn.

Hovy, Eduard H. 1993. Automated discourse generation using discourse structure relations. *Artificial Intelligence* 63: 341–385.

Hovy, Eduard H., and Chin-Yew Lin. 1999. Automated text summarization in SUMMARIST. In Mani and Maybury 1999, 81–94.

Hovy, Eduard H., and Hao Liu. 1998. Personal Communication.

Hwang, Chung Hee, and Lenhart K. Schubert. 1992. Tense trees as the "fine structure" of discourse. In *Proceedings of 30th Annual Meeting of the Association for Computational Linguistics (ACL-92)*, 232–240.

Hyland, Ken. 1998. Persuasion and context: The pragmatics of academic metadiscourse. *Journal of Pragmatics* 30(4): 437–455.

Ingwersen, Peter. 1996. Cognitive perspectives of information retrieval interaction: Elements of a cognitive IR theory. *Journal of Documentation* 52: 3–50.

InXight. 1999. `http://www.inxight.com/Products/Enterprise/SummServ.html`.

ISI. 1999. Institute for Scientific Information, `http://www.isinet.com/products/citation/citssci.html`.

ISO. 1976. Documentation—Abstracts for Publication and Documentation. ISO 214-1976. Technical report, International Organisation for Standardisation.

Iwanska, L. 1985. Discourse Structure in Factual Reports. Technical report, GE Artificial Intelligence Laboratory, NY. Unpublished.

Johnson, Frances C., Chris D. Paice, William J. Black, and A. P. Neal. 1993. The application of linguistic processing to automatic abstract generation. *Journal of Document and Text Management* 1(3): 215–241.

Jordan, M. P. 1984. *Rhetoric of Everyday English Texts*. London, UK: George Allen and Unwin.

Jurafsky, Daniel, Elizabeth Shriberg, and Debra Biasca. 1997. Switchboard SWBD-DAMSL Shallow-Discourse-Function Annotation Coders Manual. TR–97-02. Technical report, Institute of Cognitive Science, University of Colorado at Boulder, Boulder, CO.

Kan, Min-Yen, Judith L. Klavans, and Kathleen R. McKeown. 1998. Linear Segmentation and Segment Significance. In *Proceedings of the Sixth Workshop on Very Large Corpora (COLIN G/ACL-98)*, 197–205.

Kando, Noriko. 1997. Text-level structure of research papers: Implications for text-based information processing systems. In *Proceedings of BCS-IRSG Colloquium*, 68–81. Also available from `http://www.rd.nacsis.ac.jp/~kando/kando.ps`.

Kessler, Myer Mike. 1963. Bibliographic coupling between scientific papers. *American Documentation* 14(1): 10–25.

Kilgarriff, Adam. 1999. 95% replicability for manual word sense tagging. In *Proceedings of the 8th Meeting of the European Chapter of the Association for Computational Linguistics (EACL-99), Poster Session*, 277.

Kintsch, Walter, and Teun A. van Dijk. 1978. Toward a model of text comprehension and production. *Psychological Review* 85(5): 363–394.

Kircz, Joost G. 1991. The rhetorical structure of scientific articles: the case for argumentational analysis in information retrieval. *Journal of Documentation* 47(4): 354–372.

Kircz, Joost G. 1998. Modularity: The next form of scientific information presentation? *Journal of Documentation* 54: 210–235.

Klavans, Judith L., and Min-Yen Kan. 1998. Role of verbs in document analysis. In *Proceedings of 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics (ACL/COLING-98)*, 680–686.

Klavans, Judith L., Kathleen R. McKeown, and Susan Lee. 1998. Resources for evaluation of summarization techniques. In *Proceedings of First International Conference on Language Resources and Evaluation*.

Kleinberg, Jon. 1998. Authoritative sources in a hyperlinked environment. In *Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms*. Also available from `http://www.cs.cornell.edu/home/kleinber/`.

Knott, Alistair. 1996. A Data-Driven Methodology for Motivating a Set of Discourse Relations. Ph.D. thesis, Centre for Cognitive Science, University of Edinburgh, Edinburgh, UK.

Kozima, Hideki. 1993. Text segmentation based on similarity between words. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics (ACL-93)*, 286–288.

Krippendorff, Klaus. 1980. *Content Analysis: An Introduction to its Methodology*. Beverly Hills, CA: Sage Publications.

Krohn, Uwe. 1995. Visualization of navigational retrieval in virtual information spaces. In *Proceedings of the Workshop on New Paradigms in Information Visualization and Manipulation*, 26–32.

Kupiec, Julian, Jan O. Pedersen, and Francine Chen. 1995. A trainable document summarizer. In *Proceedings of the 18th Annual International Conference on Research and Development in Information Retrieval (SIGIR-95)*, 68–73.

Lancaster, Frederick Wilfrid. 1998. *Indexing and Abstracting in Theory and Practice*. London, UK: Library Association.

Lannon, John M. 1993. *Technical Writing*. New York, NY: HarperCollins Publishers, 6th edn.

Latex2Html. 1999. `http://cbl.leeds.ac.uk/nikos/tex2html/doc/latex2html/latex2html.html`.

Latour, Bruno, and Steven Woolgar. 1986. *Laboratory Life: The Social Construction of Scientific Facts*. Beverley Hills, CA: Sage Publications.

Lawrence, Steve, C. Lee Giles, and Kurt Bollacker. 1999. Digital libraries and autonomous citation indexing. *IEEE Computer* 32(6): 67–71.

Leech, Geoffrey. 1992. Corpora and theories of linguistic performance. In J. Svartvik, ed., *Directions in Corpus Linguistics*, 105–122. Berlin: Mouton de Gruyter.

Levin, Beth. 1993. *English Verb Classes and Alternations*. Chicago, IL: University of Chicago Press.

Levy, D. M. 1997. I read the news today, oh boy: Reading and attention in digital libraries. In *Proceedings of Digital Libraries T97, ACM*, 228–235.

Liddy, Elizabeth DuRoss. 1991. The discourse-level structure of empirical abstracts:

An exploratory study. *Information Processing and Management* 27(1): 55–81.

Litman, Diane J. 1996. Cue phrase classification using machine learning. *Journal of Artificial Intelligence Research* 5: 53–94.

Longacre, Robert E. 1979. The paragraph as a grammatical unit. In Talmy Givon, ed., *Syntax and Semantics: Discourse and Syntax*, vol. 12, 115–134. New York NY: Academic Press.

Luhn, Hans Peter. 1958. The automatic creation of literature abstracts. *IBM Journal of Research and Development* 2(2): 159–165.

Luukkonen, Terttu. 1992. Is scientists' publishing behaviour reward-seeking? *Scientometrics* 24: 297–319.

MacRoberts, Michael H., and Barbara R. MacRoberts. 1984. The negational reference: Or the art of dissembling. *Social Studies of Science* 14: 91–94.

Maier, Elisabeth, and Eduard H. Hovy. 1993. Organizing discourse structure relations using metafunctions. In H. Horacek and M. Zock, eds., *New Concepts in Natural Language Generation: Planning, Realization, and Systems*, 69–86. London, UK: Pinter.

Maizell, R. E., J. F. Smith, and T. E. R. Singer. 1971. *Abstracting Scientific and Technical Literature: An Introductory Guide and Texts for Scientists, Abstractors and Management*. New York, NY: Wiley-Interscience.

Malcolm, L. 1987. What rules govern tense usage in scientific articles? *English for Specific Purposes* 6: 31–43.

Mani, Inderjeet, and Eric Bloedorn. 1998. Machine learning of generic and user-focused summarization. In *Proceedings of the Fifteenth National Conference on AI (AAAI-98)*, 821–826.

Mani, Inderjeet, and Mark T. Maybury, eds. 1997. *Proceedings of the ACL/EACL-97 Workshop on Intelligent Scalable Text Summarization*.

Mani, Inderjeet, and Mark T. Maybury, eds. 1999. *Advances in Automatic Text Summarization*. Cambridge, MA: MIT Press.

Mann, William C., and Sandra A. Thompson. 1987. Rhetorical Structure Theory: A Theory of Text Organisation. ISI/RS-87-190. Technical report, Information Sciences Institute, University of Southern California, Marina del Rey, CA.

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

Manning, Alan D. 1990. Abstracts in relation to larger and smaller discourse structures. *Journal of Technical Writing and Communication* 20(4): 369–390.

Manning, Christopher D., and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. Cambridge, MA: MIT Press.

Marcu, Daniel. 1997a. From discourse structures to text summaries. In Mani and Maybury 1997, 82–88.

Marcu, Daniel. 1997b. The Rhetorical Parsing, Summarization, and Generation of Natural Language Texts. Ph.D. thesis, University of Toronto, Ont., Canada.

Marcu, Daniel. 1999a. A decision-based approach to rhetorical parsing. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL-99)*, 365–372.

Marcu, Daniel. 1999b. Discourse structures are good indicators of importance in text. In Mani and Maybury 1999, 123–136.

Maron, M. E., and J. L. Kuhns. 1960. On relevance, probabilistic indexing and information retrieval. *Journal of the Association for Computing Machinery* 7: 216–244.

Mathes, John C., and Dwight W. Stevenson. 1976. *Designing Technical Reports—Writing for Audiences in Organizations*. Indianapolis, IN: Bobbs-Merrill Educational Publishing.

Mauldin, Michael L. 1991. Retrieval performance in FERRET: A conceptual information retrieval system. In *Proceedings of the 14th Annual International Conference on Research and Development in Information Retrieval (SIGIR-91)*, 347–355.

McGirr, Clinton J. 1973. Guidelines for abstracting. *Technical Communication* 25(2): 2–5.

McKeown, Kathleen R., Karen Kukich, and James Shaw. 1994. Practical issues in automatic document generation. In *Proceedings of ANLP-94 (Applied Natural Language Processing)*, 7–14.

McKeown, Kathleen R., and Dragomir R. Radev. 1995. Generating summaries of multiple news articles. In *Proceedings of the 18th Annual International Conference on Research and Development in Information Retrieval (SIGIR-95)*, 74–82.

Michaelson, Herbert B. 1980. *How to Write and Publish Engineering Papers and Reports*. Phoenix, AZ: Oryx Press.

Microsoft. 1997. Office-97. `http://www.microsoft.com/Office/`.

Miike, Seijii, Etsuo Itoh, Kenji Ono, and Kazuo Sumita. 1994. A full-text retrieval system with a dynamic abstract generation function. In *Proceedings of the 17th Annual International Conference on Research and Development in Information Retrieval (SIGIR-94)*, 152–163.

Mikheev, Andrei. To Appear. Feature Lattices and Maximum Entropy Models. *Journal of Machine Learning* Available from `http://www.ltg.ed.ac.uk/~mikheev/papers.html`.

Mikheev, Andrei, Claire Grover, and Marc Moens. 1998. Description of the LTG system used for MUC-7. In MUC-7 1998.

Milas-Bracovic, Milica. 1987. The structure of scientific papers and their author abstracts. *Informatologia Yugoslavica* 19(1–2): 51–67.

Minel, Jean-Luc, Sylvaine Nugier, and Gerald Piat. 1997. How to appreciate the quality of automatic text summarization. In Mani and Maybury 1997, 25–30.

Minsky, M. 1975. A framework for representing knowledge. In P. Winston, ed., *The Psychology of Computer Vision*. New York, NY: McGraw-Hill.

Mitchell, John Howard. 1968. *Writing for Professional and Technical Journals*. New York, NY: J. Wiley.

Moore, Johanna D., and Cecile L. Paris. 1993. Planning text for advisory dialogues: Capturing intentional and rhetorical information. *Computational Linguistics* 19: 651–694.

Moravcsik, Michael J., and Poovanalingan Murugesan. 1975. Some results on the function and quality of citations. *Social Studies of Science* 5: 88–91.

Morris, Andrew H., George M. Kasper, and Dennis A. Adams. 1992. The effects and limitations of automated text condensing on reading comprehension performance. *Information Systems Research* 3(1): 17–35.

Morris, Jane, and Graeme Hirst. 1991. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics* 17: 21–48.

Moser, Megan G., and Johanna D. Moore. 1996. Toward a synthesis of two accounts of discourse structure. *Computational Linguistics* 22(3): 409–420.

MUC-7. 1998. *Proceedings of the Seventh Message Understanding Conference*. DARPA. www.muc.saic.com/proceedings/muc_7_toc.html.

Mullins, Nicholas C., William E. Snizek, and Kay Oehler. 1988. The structural analysis of a scientific paper. In A. F. J. van Raan, ed., *Handbook of Quantitative Studies of Science and Technology*, 81–106. Amsterdam, NL: North-Holland.

Myers, Greg. 1992. In this paper we report...—speech acts and scientific facts. *Journal of Pragmatics* 17(4): 295–313.

Nanba, Hidetsugu, and Manabu Okumura. 1999. Towards multi-paper summarization using reference information. In *Proceedings of IJCAI-99*, 926–931. http://galaga.jaist.ac.jp:8000/~nanba/study/papers.html.

Nowell, Lucy Terry, Robert K. France, Deborah Hix, Lenwood S. Heath, and Edward A. Fox. 1996. Visualizing search result: Some alternatives to query-document similarity. In *Proceedings of the 19th Annual International Conference on Research and Development in Information Retrieval (SIGIR-96)*, 67–75.

Oakes, Michael, and Chris Paice. 1999. The automatic generation of templates for automatic abstracting. In *Proceedings of the 21st BCS IRSG Colloquium on IR*.

O'Connor, John. 1982. Citing statements: Computer recognition and use to improve retrieval. *Information Processing and Management* 18(3): 125–131.

Oddy, Robert Norman, Elizabeth DuRoss Liddy, B. Balakrichnan, A. Bishop, J. Elewononi, and E. Martin. 1992. Towards the use of situational information in information retrieval. *Journal of Documentation* 48: 123–171.

O'Hara, Kenton, and Abigail Sellen. 1997. A comparison of reading paper and on-line documents. In *Proceedings of CHI-97, ACM*, 335–342. Available from http://www1.acm.org/sigchi/chi97/proceedings/paper/koh.htm.

O'Hara, Kenton, F. Smith, W. Newman, and Abigail Sellen. 1998. Student reader's use of library documents: implications for library technologies. In *Proceedings of CHI-98, ACM*, 233–240.

Olsen, Kai A., Robert R. Korfhage, Kenneth M. Sochats, Michael B. Spring, and James G. Williams. 1993. Visualizing of a document collection: The VIBE system. *Information Processing and Management* 29(1): 69–81.

Ono, Kenji, Kazuo Sumita, and Seijii Miike. 1994. Abstract generation based on rhetorical structure extraction. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING-94)*, 344–348.

Oppenheim, Charles, and Susan P. Renn. 1978. Highly cited old papers and the reasons why they continue to be cited. *Journal of the American Society for Information Science* 29: 226–230.

Oracle. 1993. Introduction to Oracle ConText. Technical report, Oracle Corporation, Redwood Shores, CA.

Paice, Chris D. 1981. The automatic generation of literary abstracts: an approach based on the identification of self-indicating phrases. In Robert Norman Oddy, Stephen E. Robertson, Cornelis Joost van Rijsbergen, and P. W. Williams, eds., *Information Retrieval Research*, 172–191. London, UK: Butterworth.

Paice, Chris D. 1990. Constructing literature abstracts by computer: techniques and prospects. *Information Processing and Management* 26: 171–186.

Paice, Chris D., and A. Paul Jones. 1993. The identification of important concepts in highly structured technical papers. In *Proceedings of the 16th Annual International Conference on Research and Development in Information Retrieval (SIGIR-93)*, 69–78.

Paris, Cecile L. 1988. Tailoring Object Descriptions to a User's Level of Expertise. *Computational Linguistics* 14(3): 64–78. Special Issue on User Modelling.

Paris, Cecile L. 1993. Dagstuhl Seminar on Summarization webpage. `http://www.ik.fh-hannover.de/ik/projekte/Dagstuhl/Abstract/Answers/Pari%s/paris.html`.

Paris, Cecile L. 1994. User Modeling in Text Generation. *Computational Linguistics* 20(2): 318–321.

Perelman, Chaim, and Lucie Olbrechts-Tyteca. 1969. *The New Rhetoric, a Tractise on Argumentation*. Notre Dame, IN: University of Notre Dame Press.

Pinelli, Thomas E., Virginia M. Cordle, and Raymond F. Vondran. 1984. The function of report components in the screening and reading of technical reports. *Journal of Technical Writing and Communication* 14(2): 87–94.

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

Pollack, Martha E. 1986. A model of plan inference that distinguishes between the beliefs of actors and observers. In *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics (ACL-86)*, 207–214.

Pollock, Joseph J., and Antonio Zamora. 1975. Automatic abstracting research at the Chemical Abstracts service. *Journal of Chemical Information and Computer Sciences* 15(4): 226–232.

Radev, Dragomir R., and Eduard H. Hovy, eds. 1998. *Working Notes of the AAAI Spring Symposium on Intelligent Text Summarization*.

Radev, Dragomir R., and Kathleen R. McKeown. 1998. Generating natural language summaries from multiple on-line sources. *Computational Linguistics* 24(3): 469–500.

Rath, G.J, A. Resnick, and T. R. Savage. 1961. The formation of abstracts by the selection of sentences. *American Documentation* 12(2): 139–143.

Raynar, Jeffrey C. 1999. Statistical models for topic segmentation. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL-99)*, 357–364.

Reed, Chris. 1999. The role of saliency in generating natural language arguments. In *Proceedings of IJCAI-99*, 876–881.

Reed, Chris, and Derek Long. 1998. Generating the structure of an argument. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics (ACL/COLING-98)*, 1091–1097.

Rees, Alan M. 1966. The relevance of relevance to the testing and evaluation of document retrieval systems. *Aslib Proceedings* 18: 316–324.

Rennie, D., and R. M. Glass. 1991. Structuring abstracts to make them more informative. *Journal of the American Medical Association* 266(1): 116–117.

Richmond, Korin, Andrew Smith, and Einat Amitay. 1997. Detecting subject boundaries within text: A language independent statistical approach. In *The Second Conference on Empirical Methods in Natural Language Processing (EMNLP-2)*.

Riley, Kathryn. 1991. Passive voice and rhetorical role in scientific writing. *Journal of Technical Writing and Communication* 21(3): 239–257.

Robertson, Stephen E., Steve Walker, Micheline M. Hancock-Beaulieu, Aaron Gull, and Marianna Lau. 1993. Okapi at TREC. In D. K. Harman, ed., *The first Text REtrieval Conference (TREC-1)*, 21–30.

Robin, Jacques. 1994. Revision-Based Generation of Natural Language Summaries Providing Historical Background. Ph.D. thesis, Computer Science Department, Columbia University, New York, NY.

Robin, Jacques, and Kathleen R. McKeown. 1996. Empirically designing and evaluating a new revision-based model for summary generation. *Artificial Intelligence* 85: 135–179.

Rowley, Jennifer. 1982. *Abstracting and Indexing*. London, UK: Bingley.

Salager-Meyer, Francoise. 1990. Discoursal flaws in medical English abstracts: A genre analysis per research- and text type. *Text* 10(4): 365–384.

Salager-Meyer, Francoise. 1991. Medical English abstracts: How well structured are they? *Journal of the American Society for Information Science* 42: 528–532.

Salager-Meyer, Francoise. 1992. A text-type and move analysis study of verb tense and modality distributions in medical English abstracts. *English for Specific Purposes* 11: 93–113.

Salager-Meyer, Francoise. 1994. Hedges and textual communicative function in medical English written discourse. *English for Specific Purposes* 13(2): 149–170.

Salton, Gerard. 1971. Cluster search strategies and the optimization of retrieval effectiveness. In Gerard Salton, ed., *The SMART Retrieval System; Experiments in Automatic Document Processing*, 223–242. Englewood Cliffs, NJ: Prentice Hill.

Salton, Gerard, James Allan, Chris Buckley, and Amit Singhal. 1994a. Automatic analysis, theme generation, and summarisation of machine readable texts. *Science* 264: 1421–1426.

Salton, Gerard, and Michael J. McGill. 1983. *Introduction to Modern Information Retrieval*. Tokyo: McGraw-Hill.

Salton, Gerard, Amit Singhal, Chris Buckley, and Mandar Mitra. 1994b. Automatic Text Decomposition Using Text Segments and Text Themes. Technical report, Cornell University.

Samuel, Ken, Sandra Carberry, and K. Vijay-Shanker. 1998. Dialogue act tagging with transformation-based learning. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics (ACL/COLING-98)*, 1150–1156.

Samuel, Ken, Sandra Carberry, and K. Vijay-Shanker. 1999. Automatically selecting useful phrases for dialogue act tagging. In *Proceedings of the Pacific Association for Computational Linguistics (PACLING-99)*.

Samuels, S. J., R. Tennyson, L. Sax, P. Mulcahy, N. Schermer, and H. Hajovy. 1987. Adults' use of text structure in the recall of a scientific journal article. *Journal of Education Research* 81: 171–174.

Saracevic, Tefko. 1975. Relevance: A review of and a framework for the thinking on the notion in information science. *Journal of the American Society for Information Science* 26(6): 321–343.

Saracevic, Tefko, Paul B. Kantor, A. Y. Chamis, and D. Trivison. 1988. A study of information seeking and retrieving. I. Background and methodology. *Journal of the American Society for Information Science* 39(3): 161–176.

Schamber, Linda, Michael B. Eisenberg, and Michael S. Nilan. 1990. A re-examination of relevance: Toward a dynamic, situational definition. *Information Processing and Management* 26: 755–776.

Schank, Roger C., and Robert P. Abelson. 1977. *Scripts, Goals, Plans and Understanding*. Hillsdale, NJ: Lawrence Erlbaum.

Schütze, Hinrich. 1998. Automatic word sense discrimination. *Computational Linguistics* 24(1): 97–124.

Sherrard, Carol. 1985. The psychology of summary writing. *Journal of Technical Writing and Communication* 15(3): 247–258.

Shum, Simon Buckingham. 1998. Evolving the web for scientific knowledge: First steps towards an "HCI knowledge web". *Interfaces, British HCI Group Magazine* 39: 16–21. Also available from `http://kmi.open.ac.uk/sbs/hciweb/Interfaces98.html`.

Shum, Simon Buckingham, Enrico Motta, and John Domingue. 1999. Representing scholarly claims in internet digital libraries: A knowledge modelling approach. In *Proceedings of ECDL'99: Third European Conference on Research and Advanced Technology for Digital Libraries*, Lecture Notes in Computer Science. Heidelberg, Germany: Springer Verlag.

Siegel, Sidney, and N. John Jr. Castellan. 1988. *Nonparametric Statistics for the Behavioral Sciences*. Berkeley, CA: McGraw-Hill, 2nd edn.

SIGMOD. 1999. `http://www.acm.org/sigs/sigmod/sigmod99`.

Sillince, John Anthony Arthur. 1992. Literature searching with unclear objectives: A new approach using argumentation. *Online Review* 16(6): 391–410.

Skorochod'ko, E. F. 1972. Adaptive method of automatic abstracting and indexing. In *Information Processing 71*, vol. 2, 1179–1182. North-Holland.

Small, Henry G. 1973. Co-citation in the scientific literature: A new measure of the relationship between two documents. *Journal of the American Society for Information Science* 24: 265–269.

Solov'ev, V. I. 1981. Functional characteristics of the author's abstract of a dissertation and the specifics of writing it. *Scientific and Technical Information Processing* 3: 80–88. English translation of *Nauchno-Tekhnicheskaya Informatsiya*, Seriya 1, Number 6, 1981, 20–24.

Spärck Jones, Karen. 1988. Tailoring Output to the User: What does User Modelling in Generation Mean? TR–158. Technical report, Computer Laboratory, University of Cambridge, Cambridge, UK.

Spärck Jones, Karen. 1990. What sort of thing is an AI experiment? In D. Partridge and Yorick Wilks, eds., *The Foundations of Artificial Intelligence: A Sourcebook*. Cambridge, UK: Cambridge University Press.

Spärck Jones, Karen. 1994. Discourse Modelling for Automatic Summarising, TR-290. Technical report, Computer Laboratory, University of Cambridge.

Spärck Jones, Karen. 1999. Automatic summarising: Factors and directions. In Mani and Maybury 1999, 1–12.

Spiegel-Rüsing, Ina. 1977. Bibliometric and content analysis. *Social Studies of Science* 7: 97–113.

Starck, Heather A. 1988. What do paragraph markings do? *Discourse Processes* 11(3): 275–304.

Strzalkowski, Tomek, Gees Stein, Jin Wang, and Bowden Wise. 1999. A robust practical text summarizer. In Mani and Maybury 1999, 137–154.

Sumita, Kazuo, Kenji Ono, Tetsuro Chino, Teruhiko Ukita, and Shin'ya Amaro. 1992. A discourse structure analyzer for Japanese text. In *Proceedings of the International Conference on Fifth Generation Computer Systems*.

Sumner, Tamara, and Simon Buckingham Shum. 1998. From documents to discourse: Shifting conceptions of scholarly publishing. In ACM Press, ed., *Proceedings of the CHI-98 ACM*, 95–102. New York, NY.

Suppe, Frederick. 1998. The structure of a scientific paper. *Philosophy of Science* 65: 381–405.

Swales, John. 1981. Aspects of Article Introductions. Aston ESP Research Project No. 1. Technical report, The University of Aston, Birmingham, U.K.

Swales, John. 1986. Citation analysis and discourse analysis. *Applied Linguistics* 7(1): 39–56.

Swales, John. 1990. *Genre Analysis: English in Academic and Research Settings. Chapter 7: Research articles in English*, 110–176. Cambridge, UK: Cambridge University Press.

Taddio, A., T. Pain, F. F. Fassos, H. Boon, A. L. Ilersich, and Elnarson T. R. 1994. Quality of nonstructured and structured abstracts of original research articles in the *British Medical Journal*, the *Canadian Medical Association Journal* and the *Journal of the American Medical Association*. *Canadian Medical Association Journal* 150(10): 1611–1615.

Taylor, Paul, Richard Caley, Alan W. Black, and Simon King. 1999. The Edinburgh Speech Tools Library (Centre for Speech Technology Research). `http://www.cstr.ed.ac.uk/projects/speech_tools/`.

Teufel, Simone. 1998. Meta-discourse markers and problem-structuring in scientific articles. In *Proceedings of the ACL-98 Workshop on Discourse Structure and Discourse Markers*, 43–49.

Teufel, Simone, Jean Carletta, and Marc Moens. 1999. An annotation scheme for discourse-level argumentation in research articles. In *Proceedings of the 8th Meeting of the European Chapter of the Association for Computational Linguistics (EACL-99)*, 110–117.

Teufel, Simone, and Marc Moens. 1997. Sentence extraction as a classification task. In Mani and Maybury 1997, 58–65.

Teufel, Simone, and Marc Moens. 1998. Sentence extraction and rhetorical classification for flexible abstracts. In Radev and Hovy 1998, 16–25.

Teufel, Simone, and Marc Moens. 1999a. Argumentative classification of extracted sentences as a first step towards flexible abstracting. In Mani and Maybury 1999, 155–171.

Teufel, Simone, and Marc Moens. 1999b. Discourse-level argumentation in scientific articles: human and automatic annotation. In *Proceedings of ACL-99 Workshop "Towards Standards and Tools for Discourse Tagging"*, 84–93.

Teufel, Simone, and Marc Moens. In Prep. Argumentative Zoning of Scientific Text.

Thomas, Sarah, and Thomas Hawes. 1994. Reporting verbs in medical journal articles. *English for Specific Purposes* 13(4): 129–148.

Thompson, Geoff, and Ye Yiyun. 1991. Evaluation in the reporting verbs used in academic papers. *Applied Linguistics* 12(4): 365–382.

Tibbo, Helen R. 1992. Abstracting across the disciplines: A content analysis of abstracts from the natural sciences, and the humanities with implications for abstracting standards and online information retrieval. *Library and Information Science Research* 14(1): 31–56.

Tipster SUMMAC. 1999. `http://www.itl.nist.gov/div894/894.02/related_projects/tipster_summac/i%ndex/cmp_lg.html`.

Toulmin, Stephen, ed. 1972. *Human Understanding: The Collective Use and Evolution of Concepts*. Princeton, NJ: Princeton University Press.

Trawinski, Bogdan. 1989. A methodology for writing problem-structured abstracts. *Information Processing and Management* 25(6): 693–702.

van Dijk, Teun A. 1980. *Macrostructures: An Interdisciplinary Study of Global Structures in Discourse, Interaction and Cognition*. Hillsdale, NJ: Lawrence Erlbaum.

van Eemeren, F. H., R. Grootendorst, and F. Snoeck-Henkemans. 1996. *Fundamentals of Argumentation Theory*. Lawrence Erlbaum.

van Emden, Joan, and Jennifer Easteal. 1996. *Technical Writing and Speaking: An Introduction*. London, UK: McGraw-Hill.

van Rijsbergen, Cornelis Joost. 1979. *Information Retrieval*. London, UK: Butterworth, 2nd edn.

Weil, B. H., H. Owen, and I. Zarember. 1963. Technical abstracting fundamentals. II. Writing principles and practices. *Journal of Chemical Documentation* 3(2): 125–132.

Weinstock, Melvin. 1971. Citation indexes. In *Encyclopedia of Library and Information Science*, vol. 5, 16–40. New York, NY: Dekker.

Wellons, M. E., and G. P. Purcell. 1999. Task-specific extracts for using the medical literature. In *Proceedings of the American Medical Informatics Symposium*, 1004–1008.

West, Gregory K. 1980. That-nominal constructions in traditional rhetorical divisions of scientific research papers. *TESOL Quarterly* 14(4): 483–488.

Wiebe, Janyce. 1994. Tracking point of view in narrative. *Computational Linguistics* 20(2): 223–287.

Wiebe, Janyce, Rebecca F. Bruce, and Thomas P. O'Hara. 1999. Development and use of a gold-standard data set for subjectivity classifications. In *Proceedings of 37th Annual Meeting of the Association for Computational Linguistics (ACL-99)*, 246–253.

Yarowsky, David. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL-95)*, 189–196.

Yianilos, Peter. 1997. The LikeIt Intelligent String Comparison Facility. TR-97-093. Technical report, NEC Research Institute, Princeton, NJ. Also available from `http://www.neci.nj.nec.com/homepages/pny/papers/likeit/main.html`.

Zappen, James P. 1983. A rhetoric for research in sciences and technologies. In Paul V. Anderson, R. John Brockman, and Carolyn R. Miller, eds., *New Essays in Technical and Scientific Communication Research Theory Practice*, 123–138. Farmingdale, NY: Baywood Publishing Company, Inc.

Zechner, Klaus. 1995. Automatic Text Abstracting by Selecting Relevant Passages. Master's thesis, Centre for Cognitive Science, University of Edinburgh, Edinburgh, UK.

Ziman, John M. 1968. *Public Knowledge: An Essay Concerning the Social Dimensions of Science*. Cambridge, UK: Cambridge University Press.

Ziman, John M. 1969. Information, Communication, Knowledge. *Nature* 224: 318–324.

Zuckerman, Harriet, and Robert K. Merton. 1973. Institutionalized patterns of evaluation in science. In Robert K. Merton, ed., *The Sociology of Science: Theoretical and Empirical Investigations*, 460–496. Chicago, IL: University of Chicago Press.