Improving Summaries by Revising Them

Inderjeet Mani and **Barbara Gates** and **Eric Bloedorn**
The MITRE Corporation
11493 Sunset Hills Rd.
Reston, VA 22090, USA
{imani,blgates,bloedorn}@mitre.org

Abstract

This paper describes a program which revises a draft text by aggregating together descriptions of discourse entities, in addition to deleting extraneous information. In contrast to knowledge-rich sentence aggregation approaches explored in the past, this approach exploits statistical parsing and robust coreference detection. In an evaluation involving revision of topic-related summaries using informativeness measures from the TIPSTER SUMMAC evaluation, the results show gains in informativeness without compromising readability.

1 Introduction

Writing improves with revision. Authors are familiar with the process of condensing a long paper into a shorter one: this is an iterative process, with the results improved over successive drafts. Professional abstractors carry out substantial revision and editing of abstracts (Cremmins 1996). We therefore expect revision to be useful in automatic text summarization. Prior research exploring the use of revision in summarization, e.g., (Gabriel 1988), (Robin 1994), (McKeown et al. 1995) has focused mainly on structured data as the input. Here, we examine the use of revision in summarization of text input.

First, we review some summarization terminology. In revising draft summaries, these condensation operations, as well as stylistic rewording of sentences, play an important role. Summaries can be used to indicate what topics are addressed in the source text, and thus can be used to alert the user as to the source content (the *indicative* function). Summaries can also be used to cover the concepts in the source text to the extent possible given the compression requirements for the summary (the *informative* function). Summaries can be tailored to a reader's interests and expertise, yielding *topic-related* summaries, or they can be aimed at a particular - usually broad - readership community, as in the case of (so-called) *generic* summaries. Revision here applies to generic and topic-related informative summaries, intended for publishing and dissemination.

Summarization can be viewed as a text-to-text reduction operation involving three main condensation operations: selection of salient portions of the text, aggregation of information from different portions of the text, and abstraction of specific information with more general information (Mani and Maybury 1999). Our approach to revision is to construct an initial draft summary of a source text and then to add to the draft additional background information. Rather than concatenate material in the draft (as surface-oriented, sentence extraction summarizers do), information in the draft is combined and excised based on revision rules involving aggregation (Dalianis and Hovy 1996) and elimination operations. Elimination can increase the amount of compression (summary length/source length) available, while aggregation can potentially gather and draw in relevant background information, in the form of descriptions of discourse entities from different parts of the source. We therefore hypothesize that these operations can result in packing in more information per unit compression than possible by concatenation. Rather than opportunistically adding as much background information that can fit in the available compression, as in (Robin 1994), our approach adds background information from the source text to the draft based on an information weighting function.

Our revision approach assumes input sentences are represented as syntactic trees whose

nodes are annotated with coreference information. In order to provide open-domain coverage the approach does not assume a meaning-level representation of each sentence, and so, unlike many generation systems, the system does not represent and reason about what is being said[1]. Meaning-dependent revision operations are restricted to situations where it is clear from coreference that the same entity is being talked about.

There are several criteria our revision model needs to satisfy. The final draft needs to be *informative, coherent*, and *grammatically well-formed.* Informativeness is explored in Section 4.2. We can also strive to guarantee, based on our revision rule set, that each revision will be syntactically well-formed. Regarding coherence, revision alters rhetorical structure in a way which can produce disfluencies. As rhetorical structure is hard to extract from the source[2], our program instead uses coreference to guide the revision, and attempts to patch the coherence by adjusting references in revised drafts.

2 The Revision Program

The summary revision program takes as input a source document, a draft summary specification, and a target compression rate. Using revision rules, it generates a revised summary draft whose compression rate is no more than δ above the target compression rate. The initial draft summary (and background) are specified in terms of a task-dependent weighting function which indicates the relative importance of each of the source document sentences. The program repeatedly selects the highest weighted sentence from the source and adds it to the initial draft until the given compression percentage of the source has been extracted, rounded to the nearest sentence. Next, for each rule in the sequence of revision rules, the program repeatedly applies the rule until it can no longer be applied. Each rule application results in a revised draft. The program selects sentences for rule application by giving preference to higher weighted sentences. A unary rule applies to a single sentence. A binary rule applies to a pair of sentences, at least one of which must be in the draft, and where the first sentence precedes the second in the input. Control over sentence complexity is imposed by failing rule application when the draft sentence is too long, the parse tree is too deep[3], or if more than two relative clauses would be stacked together. The program terminates when there are no more rules to apply or when the revised draft exceeds the required compression rate by more than δ.

The syntactic structure of each source sentence is extracted using Apple Pie 7.2 (Sekine 1998), a statistical parser trained on Penn Treebank data. It was evaluated by (Sekine 1998) as having 79% F-score accuracy (parseval) on short sentences (less than 40 words) from the Treebank. An informal assessment we made of the accuracy of the parser (based on intuitive judgments) on our own data sets of news articles suggests about 66% of the parses were acceptable, with almost half of the remaining parsing errors being due to part-of-speech tagging errors, many of which could be fixed by preprocessing the text. To establish coreference between proper names, named entities are extracted from the document, along with coreference relations using SRA's NameTag 2.0 (Krupka 1995), a MUC-6 fielded system. In addition, we implemented our own coreference extension: A singular definite NP (e.g., beginning with "the", and not marked as a proper name) is marked by our program as coreferential (i.e., in the same coreference equivalence class) with the last singular definite or singular indefinite atomic NP with the same head, provided they are within a distance γ of each other. On a corpus of 90 documents, drawn from the TIPSTER evaluation, described in Section 4.1 below, this coreference extension scored 94% precision (470 valid coreference classes/501 total coreference classes) on definite NP coreference. Also, "he" (likewise "she") is marked, subject to γ, as coreferential with the last person name mentioned, with gender agreement enforced when the person's first name's gender is known (from NameTag's list of common first names)[4]. Most

[1] Note that professional abstractors do not attempt to fully "understand" the text - often extremely technical material, but use surface-level features as above as well as the overall discourse structure of the text (Cremmins 1996).

[2] However, recent progress on this problem (Marcu 1997) is encouraging.

[3] Lengths or depths greater than two standard deviations beyond the mean are treated as too long or deep.

[4] However, this very naive method was excluded from

rule-name: rel-clause-intro-which-1
patterns:
?X1 ; # first sentence pattern
?Y1 ?Y2 ?Y3 # second sentence pattern
tests:
label-NP ?X1 ; not entity-class ?X1 person ;
label-S ?Y1 ;
root ?Y1 ;
label-NP ?Y2 ;
label-VP ?Y3 ;
adjacent-sibling ?Y2 ?Y3 ;
parent-child ?Y1 ?Y2 ;
parent-child ?Y1 ?Y3 ;
coref ?X1 ?Y2
actions:
subs ?X1 (NP ?X1 (, -COMMA-)
(SBAR (WHNP (WP which))
(S ?Y3)) (, -COMMA-)) ;
elim-root-of ?Y1 # removes second sentence

Figure 2: Relative Clause Introduction Rule showing Aggregation and Elimination operations.

of the errors were caused by different sequences of words between the determiner and the noun phrase head word (e.g., "the factory" = "the cramped five-story pre-1915 factory" is OK, but "the virus program"= "the graduate computer science program" isn't).

3 Revision Rules

The revision rules carry out three types of operations. *Elimination* operations eliminate constituents from a sentence. These include elimination of parentheticals, and sentence-initial PPs and adverbial phrases satisfying lexical tests (such as "In particular,", "Accordingly," "In conclusion," etc.)[5].

Aggregation operations combine constituents from two sentences, at least one of which must be a sentence in the draft, into a new constituent which is inserted into the draft sentence. The basis for combining sentences is that of referential identity: if there is an NP in sentence i which is coreferential with an NP in sentence j, then sentences i and j are candidates for aggregation. The most common form of aggregation is expressed as tree-adjunction (Joshi 1998) (Dras 1999). Figures 1 and 2 show a relative clause introduction rule which turns a VP of a (non-embedded) sentence whose subject is coreferential with an NP of an earlier (draft) sentence into a relative clause modifier of the draft sentence NP. Other appositive phrase insertion rules include copying and inserting nonrestrictive relative clause modifiers (e.g., "Smith, *who...*,"), appositive modifiers of proper names (e.g., "Peter G. Neumann, *a computer security expert familiar with the case*,..."), and proper name appositive modifiers of definite NPs (e.g., "The network, *named ARPANET*, is operated by ..").

Smoothing operations apply to a single sentence, performing transformations so as to arrive at more compact, stylistically preferred sentences. There are two types of smoothing. *Reduction* operations simplify coordinated constituents. Ellipsis rules include subject ellipsis, which lowers the coordination from a pair of clauses with coreferential subjects to their VPs (e.g., "The rogue computer program destroyed files over a five month period and the program infected close to 100 computers at NASA facilities" $\Longrightarrow$ "The rogue computer program destroyed files over a five month period and infected close to 100 computers at NASA facilities"). It usually applies to the result of an aggregation rule which conjoins clauses whose subjects are coreferential. Relative clause reduction includes rules which apply to clauses whose VPs begin with "be" (e.g., "which is" is deleted) or "have" (e.g., "which have" $\Longrightarrow$ "with"), as well as for other verbs, a rule deleting the relative pronoun and replacing the verb with its present participle (i.e., "which V" $\Longrightarrow$ "V+ing"). Coordination rules include relative clause coordination. *Reference Adjustment* operations fix up the results of other revision operations in order to improve discourse-level coherence, and as a result, they are run last[6]. They include substitution of a proper name with a name alias if the name is mentioned earlier, expansion of a pronoun with a coreferential proper name in a parenthetical ("pronoun expansion"), and ("indefinitization") replacement of a definite NP with a coreferential indefinite if the definite occurs without a prior indefinite.

our analysis because of a system bug.

[5]Such lexical tests help avoid misrepresenting the meaning of the sentence.

[6]Such operations have been investigated earlier by (Robin 1994).

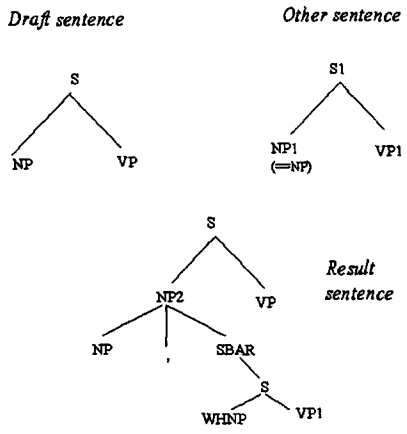

Figure 1: Relative Clause Introduction showing tree NP2 being adjoined into tree S

4 Evaluation

Evaluation of text summarization and other such NLP technologies where there may be many acceptable outputs, is a difficult task. Recently, the U.S. government conducted a large-scale evaluation of summarization systems as part of its TIPSTER text processing program (Mani et al. 1999), which included both an extrinsic (relevance assessment) evaluation, as well as an intrinsic (coverage of key ideas) evaluation. The test set used in the latter (Q&A) evaluation along with several automatically scored measures of informativeness has been reused in evaluating the informativeness of our revision component.

4.1 Background: TIPSTER Q&A Evaluation

In this Q&A evaluation, the summarization system, given a document and a topic, needed to produce an informative, topic-related summary that contained the correct answers found in that document to a set of topic-related questions. These questions covered "obligatory" information that has to be provided in any document judged relevant to the topic. The topics chosen (3 in all) were drawn from the TREC (Harman and Voorhees 1996) data sets. For each topic, 30 relevant TREC documents were chosen as the source texts for topic-related summarization. The principal tasks of each Q&A evaluator were to prepare the questions and answer keys and to score the system summaries. To construct the answer key, each evaluator marked off any passages in the text that provided an answer to a question (example shown in Table 1).

Two kinds of scoring were carried out. In the first, a manual method, the answer to each question was judged Correct, Partially Correct, or Missing based on guidelines involving a human comparison of the summary of a document against the set of tagged passages for that question in the answer key for that document. The second method of scoring was an automatic method. This program[7] took as input a key file and a summary to be scored, and returns an informativeness score on four different metrics. The key file includes tags identifying passages in the file which answer certain questions. The scoring uses the overlap measures shown in Table 2[8]. The automatically computed V4 thru V7 informativeness scores were strongly correlated with the human-evaluated scores (Pearson $r > .97$, $\alpha < 0.0001$). Given this correlation, we decided to use these informativeness measures.

4.2 Revision Evaluation: Informativeness

To evaluate the revised summaries, we first converted each summary into a weighting function which scored each full-text sentence in the summary's source in terms of its similarity to the most similar summary sentence. The weight of a source document sentence s given a sum-

[7] The program was reimplemented by us for use in the revision evaluation.

[8] Passage matching here involves a sequential match with stop words and punctuation removed.

Title : Computer Security
Description : Identify instances of illegal entry into sensitive computer networks by nonauthorized personnel.
Narrative : Illegal entry into sensitive computer networks is a serious and potentially menacing problem. Both 'hackers' and foreign agents have been known to acquire unauthorized entry into various networks. Items relative this subject would include but not be limited to instances of illegally entering networks containing information of a sensitive nature to specific countries, such as defense or technology information, international banking, etc. Items of a personal nature (e.g. credit card fraud, changing of college test scores) should not be considered relevant.
Questions
1)Who is the known or suspected hacker accessing a sensitive computer or computer network?
2) How is the hacking accomplished or putatively achieved?
3) Who is the apparent target of the hacker?
4) What did the hacker accomplish once the violation occurred? What was the purpose in performing the violation?
5) What is the time period over which the breakins were occurring?

```
As a federal grand jury decides whether he should be prosecuted, <Q1>a graduate
student</Q1> linked to a ''virus'' that disrupted computers nationwide <Q5>last
month</Q5>has been teaching his lawyer about the technical subject and turning down
offers for his life story. ....No charges have been filed against <Q1>Morris</Q1>,
who reportedly told friends that he designed the virus that temporarily clogged about
<Q3>6,000 university and military computers</Q3> <Q2>linked to the Pentagon's Arpanet
network</Q2>. .....
```

Table 1: Q&A Topic 258, topic-related questions, and part of a relevant source document showing answer key annotations.

Overlap Metric	Definition
V4	full credit if the text spans for all tagged key passages are found in their entirety in the summary
V5	full credit if the text spans for all tagged key passages are found in their entirety in the summary; half credit if the text spans for all tagged key passages are found in some combination of full or truncated form in the summary
V6	full credit if the text spans for all tagged key passages are found in some combination of full or truncated form in the summary
V7	percentage of credit assigned that is commensurate with the extent to which the text spans for tagged key passages are present in the summary

Table 2: Informativeness measures for Automatic Scoring of each question that has an answer according to the key.

Party	FOG		Kincaid	
	Before	After	Before	After
CGI/CMU	16.49	15.50	13.22	12.23
Cornell/SabIR	15.51	15.08	12.15	11.71
GE	15.43	15.14	12.13	11.87
ISI	19.57	17.94	16.18	14.51
NMSU	16.54	15.52	13.32	12.30
SRA	15.59	15.29	12.26	11.99
UPenn	16.29	16.21	12.93	12.83
Mean	16.48	15.82	13.15	12.51

Table 3: Readability of Summaries Before (Original Summary) and After Revision (A+E). Overall, both FOG and Kincaid scores show a slight but statistically significant drop on revision ($\alpha < 0.05$).

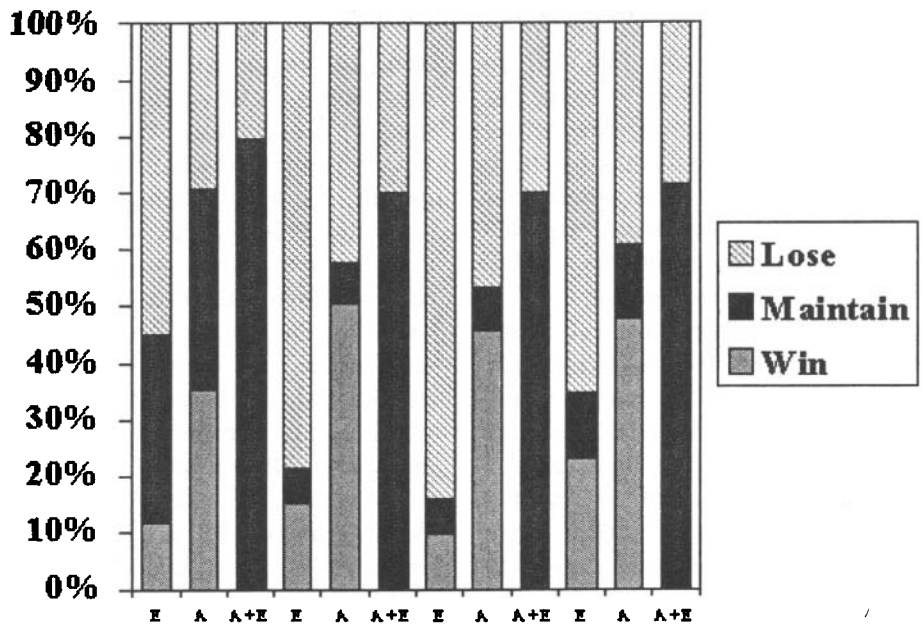

Figure 3: Gains in Compression-Normalized Informativeness of revised summaries compared to initial drafts. E = elimination, A = aggregation. A, E, and A+E are shown in the order V4, V5, V6, and V7.

<S1> Researchers today tried to trace a "virus" that infected computer systems nationwide, <Q4> slowing machines in universities, a NASA and nuclear weapons lab and other federal research centers linked by a Defense Department computer network. </Q4> <S3> Authorities said the virus , **which** <FROM S16> <Q3> *the virus* **infected only unclassified computers** </Q3> **and** <FROM S15> <Q3> *the virus* **affected the unclassified, non-secured computer systems** </Q3> **(and** *which* <FROM S19> <Q4> *the virus was* **"mainly just** *slowing down* **systems)** *and slowing data "*, </Q4> apparently <Q4> destroyed no data but temporarily halted some research. </Q4> <S14>. The computer problem also was discovered late Wednesday at the <Q3> Lawrence Livermore National Laboratory in Livermore, Calif. </Q3> *<S15>* <S20> "The developer was clearly a very high order hacker,", <FROM S25> <Q1> **a graduate student** </Q1> <Q2> *who made* **making a programming error in designing the virus,***causing the program to replicate faster than expected* </Q2> *or computer buff, said John McAfee, chairman of the Computer Virus Industry Association in Santa Clara, Calif..* <S24> **The Times reported today that** *the anonymous caller* **an anonymous caller to the paper said his associate was responsible for the attack and had meant it to be harmless.**

Figure 4: A revised summary specified in terms of an original draft (plain text) with added (boldface) and deleted (italics) spans. Sentence <S> and Answer Key <Q> tags are overlaid.

mary is the match score of s's best-matching summary sentence, where the match score is the percentage of content word occurrences in s that are also found in the summary sentence. Thus, we constructed an idealized model of each summary as a sentence extraction function. Since some of the participants truncated and occasionally mangled the source text (in addition, Penn carried out pronoun expansion), we wanted to avoid having to parse and apply revision rules to such relatively ill-formed material. This idealization is highly appropriate, for each of the summarizers considered[9] did carry out sentence extraction; in addition, it helps level the playing field, avoiding penalization of individual summarizers simply because we didn't cater to the particular form of their summary.

Each summary was revised by calling the revision program with the full-text source, the original compression rate of the summary, and

[9]TextWise, which extracted named entities rather than passages, was excluded.

the summary weighting function (i.e., with the weight for each source sentence). The 630 revised summaries (3 topics × 30 documents per topic × 7 participant summaries per document) were then scored against the answer keys using the overlap measures above. The documents consisted of AP, Wall Street Journal, and Financial Times news articles from the TREC (Harman and Voorhees 1996) collection.

The rules used in the system are very general, and were not modified for the evaluation except for turning off most of the reference adjustment rules, as we wished to evaluate that component separately. Since the answer keys typically do not contain names of commentators, we wanted to focus the algorithm away from such names (otherwise, it would aggregate information around those commentators). As a result, special rules were written in the revision rule language to detect commentator names in reported speech ("X said that ..", "X said ...", ", said X..", ", said X..", etc.), and these names were added to a stoplist for use in entityhood and coreference tests during regular revision rule application.

Figure 3 shows percentage of losses, maintains, and wins in informativeness against the initial draft (i.e., the result of applying the compression to the sentence weighting function). Informativeness using V7 is measured by V7[10] normalized for compression as:

$$nV7 = V7 * (1 - \frac{s1}{s0}) \quad (1)$$

where $s1$ is summary length and $s0$ is the source length. This initial draft is in itself not as informative as the original summary: in all cases except for Penn on 257, the initial draft either maintains or loses informativeness compared to the original summary.

As Figure 3 reveals (e.g., for nV7), revising the initial draft using elimination rules only (E) results in summaries which are less informative than the initial draft 65% of the time, suggesting that these rules are removing informative material. Revising the initial draft using aggregation rules alone (A), by contrast, results in more informative summaries 47% of the time, and equally informative summaries another 13% of the time. This is due to aggregation folding in additional informative material into the initial draft when it can. Inspection of the output summaries, an example of which is shown in Figure 4, confirms the folding in behavior of aggregation. Finally, revising the initial draft using both aggregation and elimination rules (A+E) does no more than maintain the informativeness of the initial draft, suggesting A and E are canceling each other out. The same trend is observing for nV4 thru nV6, confirming that the relative gain in informativeness due to aggregation is robust across a variety of (closely related) measures. Of course, if the revised summaries were instead radically different in wording from the original drafts, such informativeness measures would, perhaps, fall short.

It is also worth noting the impact of aggregation is modulated by the current control strategy; we don't know what the upper bound is on how well revision could do given other control regimes. Overall, then, while the results are hardly dramatic, they are certainly encouraging[11].

4.3 Revision Evaluation: Readability

Inspection of the results of revision indicates that the syntactic well-formedness revision criterion is satisfied to a very great extent. Improper extraction from coordinated NPs is an issue (see Figure 4), but we expect additional revision rules to handle such cases. Coherence disfluencies do occur; for example, since we don't resolve possessive pronouns or plural definites, we can get infelicitous revisions like "A computer virus, which entered **their computers* through ARPANET, infected systems from MIT." Other limitations in definite NP coreference can and do result in infelicitous reference adjustments. For one thing, we don't link definites to proper name antecedents, resulting in inappropriate indefinitization (e.g., "Bill Gates ... **A computer tycoon*"). In addition, the "same head word" test doesn't of course address inferential relationships between the definite NP and its antecedent (even when the antecedent is explicitly mentioned), again resulting in inappropriate indefinitization (e.g., "The program ... **a developer*", and "The developer

[10] V7 computes for each question the percentage of its answer passages completely covered by the summary. This normalization is extended similarly for V4 thru V6.

[11] Similar results hold while using a variety of other compression normalization metrics.

... An anonymous caller said *a very high order hacker* was a graduate student").

To measure fluency without conducting an elaborate experiment involving human judgments, we fell back on some extremely coarse measures based on word and sentence length computed by the (gnu) unix program *style* (Cherry 1981). The FOG index sums the average sentence length with the percentage of words over 3 syllables, with a "grade" level over 12 indicating difficulty for the average reader. The Kincaid index, intended for technical text, computes a weighted sum of sentence length and word length. As can be seen from Table 3, there is a slight but significant lowering of scores on both metrics, revealing that according to these metrics revision is not resulting in more complex text. This suggests that elimination rather than aggregation is mainly responsible for this.

5 Conclusion

This paper demonstrates that recent advances in information extraction and robust parsing can be exploited effectively in an open-domain model of revision inspired by work in natural language generation. In the future, instead of relying on adjustment rules for coherence, it may be useful to incorporate a level of text planning. We also hope to enrich the background information by merging information from multiple text and structured data sources.

References

Cherry, L.L., and Vesterman, W. *Writing Tools: The STYLE and DICTION programs*, Computer Science Technical Report 91, Bell Laboratories, Murray Hill, N.J. (1981).

Cremmins, E. T. 1996. *The Art of Abstracting.* Information Resources Press.

Dalianis, H., and Hov, E. 1996. *Aggregation in Natural Language Generation.* In Zock, M., and Adorni, G., eds., Trends in Natural Language Generation: an Artificial Intelligence Perspective, pp.88-105. Lecture Notes in Artificial Intelligence, Number 1036, Springer Verlag, Berlin.

Dras, M. 1999. Tree Adjoining Grammar and the Reluctant Paraphrasing of Text, Ph.D. Thesis, Macquarie University, Australia.

Gabriel, R. 1988. Deliberate Writing. In McDonald, D.D., and Bolc, L., eds., Natural Language Generation Systems, Springer-Verlag, NY.

Harman, D.K. and E.M. Voorhees. 1996. The fifth text retrieval conference (trec-5). *National Institute of Standards and Technology NIST SP 500-238.*

Joshi, A. K. and Schabes, Y. 1996. "Tree-Adjoining Grammars". In Rosenberg, G., and Salomaa, A., eds., Handbook of Formal Languages, Vol. 3, 69-123. Springer-Verlag, NY.

Krupka, G. 1995. "SRA: Description of the SRA System as Used for MUC-6", Proceedings of the Sixth Message Understanding Conference (MUC-6), Columbia, Maryland, November 1995.

Marcu, D. 1997. From discourse structures to text summaries. *in Mani, I. and Maybury, M., eds., Proceedings of the ACL/EACL'97 Workshop on Intelligent Scalable Text Summarization.*

Mani, I. and M. Maybury, eds. 1999. *Advances in Automatic Text Summarization.* MIT Press.

Mani, I., Firmin, T., House, D., Klein, G., Hirschman, L., and Sundheim, B. 1999. "The TIPSTER SUMMAC Text Summarization Evaluation", Proceedings of EACL'99, Bergen, Norway, June 8-12, 1999.

McKeown, K., J. Robin, and K. Kukich. 1995. Generating Concise Natural Language Summaries. *Information Processing and Management, 31,5, 703-733.*

Robin, J. 1994. Revision-based generation of natural language summaries providing historical background: corpus-based analysis, design and implementation. Ph.D. Thesis, Columbia University.

Sekine, S. 1998. Corpus-based Parsing and Sublanguage Studies. Ph.D. Dissertation, New York University, 1998.