# Summarization-based Query Expansion in Information Retrieval

## Tomek Strzalkowski, Jin Wang, and Bowden Wise

GE Corporate Research and Development
1 Research Circle
Niskayuna, NY 12309
strzalkowski@crd.ge.com

## Abstract

We discuss a semi-interactive approach to information retrieval which consists of two tasks performed in a sequence. First, the system assists the searcher in building a comprehensive statement of information need, using automatically generated topical summaries of sample documents. Second, the detailed statement of information need is automatically processed by a series of natural language processing routines in order to derive an optimal search query for a statistical information retrieval system. In this paper, we investigate the role of automated document summarization in building effective search statements. We also discuss the results of latest evaluation of our system at the annual Text Retrieval Conference (TREC).

## Information Retrieval

Information retrieval (IR) is a task of selecting documents from a database in response to a user's query, and ranking them according to relevance. This has been usually accomplished using statistical methods (often coupled with manual encoding) that (a) select terms (words, phrases, and other units) from documents that are deemed to best represent their content, and (b) create an inverted index file (or files) that provide an easy access to documents containing these terms. A subsequent search process attempts to match preprocessed user queries against term-based representations of documents in each case determining a degree of relevance between the two which depends upon the number and types of matching terms.

A search is successful if it can return as many as possible documents which are *relevant* to the query, with as few as possible non-relevant documents. In addition, the relevant documents should be ranked ahead of non-relevant ones. The quantitative text representation methods, predominant in today's leading information retrieval systems[1] limit

---

[1] Representations anchored on words, word or char-

the system's ability to generate a successful search because they rely more on the *form* of a query than on its *content* in finding document matches. This problem is particularly acute in *ad-hoc* retrieval situations where the user has only a limited knowledge of database composition and needs to resort to generic or otherwise incomplete search statements. In order to overcome this limitation, many IR systems allow varying degrees of user interaction that facilitates query optimization and calibration to closer match user's information seeking goals. A popular technique here is relevance feedback, where the user or the system judges the relevance of a sample of results returned from an initial search, and the query is subsequently rebuilt to reflect this information. Automatic relevance feedback techniques can lead to a very close mapping of *known* relevant documents, however, they also tend to overfit, which in turn reduces their ability of finding *new* documents on the same subject. Therefore, a serious challenge for information retrieval is to devise methods for building better queries, or in assisting user to do so.

## Building effective search queries

We have been experimenting with manual and automatic natural language query (or topic, in TREC parlance) building techniques. This differs from most query modification techniques used in IR in that our method is to reformulate the user's statement of information need rather than the search system's internal representation of it, as relevance feedback does. Our goal is to devise a method of full-text expansion that would allow for creating exhaustive search topics such that: (1) the performance of any system using the expanded topics would be significantly better than when the system is run using the original topics, and (2) the method of topic

---

acter sequences, or some surrogates of these, along with significance weights derived from their distribution in the database.

expansion could eventually be automated or semi-automated so as to be useful to a non-expert user. Note that the first of the above requirements effectively calls for a free text, unstructured, but highly precise and exhaustive description of user's search statement. The preliminary results from TREC evaluations show that such an approach is indeed very effective.

One way to view query expansion is to make the user query resemble more closely the documents it is expected to retrieve. This may include both content, as well as some other aspects such as composition, style, language type, etc. If the query is indeed made to resemble a "typical" relevant document, then suddenly everything about this query becomes a valid search criterion: words, collocations, phrases, various relationships, etc. Unfortunately, an average search query does not look anything like this, most of the time. It is more likely to be a statement specifying the semantic criteria of relevance. This means that except for the semantic or conceptual resemblance (which we cannot model very well as yet) much of the appearance of the query (which we can model reasonably well) may be, and often is, quite misleading for search purposes. Where can we get the right queries?

In today's information retrieval, query expansion usually is typically limited to adding, deleting or re-weighting of terms. For example, content terms from documents judged relevant are added to the query while weights of all terms are adjusted in order to reflect the relevance information. Thus, terms occurring predominantly in relevant documents will have their weights increased, while those occurring mostly in non-relevant documents will have their weights decreased. This process can be performed automatically using a relevance feedback method, e.g., (Rocchio 1971), with the relevance information either supplied manually by the user (Harman 1988), or otherwise guessed, e.g. by assuming top 10 documents relevant, etc. (Buckley, et al. 1995). A serious problem with this term-based expansion is its limited ability to capture and represent many important aspects of what makes some documents relevant to the query, including particular term co-occurrence patterns, and other hard-to-measure text features, such as discourse structure or stylistics. Additionally, relevance-feedback expansion depends on the inherently partial relevance information, which is normally unavailable, or unreliable. Other types of query expansions, including general purpose thesauri or lexical databases (e.g., Wordnet) have been found generally unsuccessful in information retrieval, (Voorhees 1994).

An alternative to term-only expansion is a full-text expansion described in (Strzalkowski et al. 1997). In this approach, search topics are expanded by pasting in entire sentences, paragraphs, and other sequences directly from any text document. To make this process efficient, an initial search is performed with the unexpanded queries and the top N (10-30) returned documents are used for query expansion. These documents, irrespective of their overall relevancy to the search topic, are scanned for passages containing concepts referred to in the query. The resulting expanded queries undergo further text processing steps, before the search is run again. We need to note that the expansion material was found in both relevant and non-relevant documents, benefiting the final query all the same. In fact, the presence of such text in otherwise non-relevant documents underscores the inherent limitations of distribution-based term reweighting used in relevance feedback.

In this paper, we describe a method of full-text topic expansion where the expansion passages are obtained from an automatic text summarizer. A preliminary examination of TREC-6 results indicate that this mode of expansion is at least as effective as the purely manual expansion which requires the users to read entire documents to select expansion passages. This brings us a step closer to a fully automated expansion: the human-decision factor has been reduced to an accept/reject decision for expanding the search query with a summary.

## Summarization-based query expansion

We used our automatic text summarizer to derive query-specific summaries of documents returned from the first round of retrieval. The summaries were usually 1 or 2 consecutive paragraphs selected from the original document text. The initial purpose was to show to the user, by the way of a quick-read abstract, why a document has been retrieved. If the summary appeared relevant and moreover captured some important aspect of relevant information, then the user had an option to paste it into the query, thus increasing the chances of a more successful subsequent search. Note again that it wasn't important if the summarized documents were themselves relevant, although they usually were.

The query expansion interaction proceeds as follows:

1. The initial natural language statement of information need is submitted to SMART-based NLIR retrieval engine via a Query Expansion Tool (QET) interface. The statement is converted into an in-

ternal search query and run against the TREC database.[2]

2. NLIR returns top N (=30) documents from the database that match the search query.

3. The user determines a topic for the summarizer. By default, it is the title field of the initial search statement (see below).

4. The summarizer is invoked to automatically summarize each of the N documents with respect to the selected topic.

5. The user reviews the summaries (spending approx. 5-15 seconds per summary) and *de-selects* these that are *not* relevant to the search statement.

6. All remaining summaries are automatically attached to the search statement.

7. The expanded search statement is passed through a series of natural language processing steps and then submitted for the final retrieval.

A partially expanded TREC Topic 304 is shown below. The original topic comprises the first four fields, with the Expanded field added through the query expansion process. The initial query, while somewhat lengthy by IR standards (though not by TREC standards) is still quite generic in form, that is, it supplies few specifics to guide the search. In contrast, the Expanded section supplies not only many concrete examples of relevant concepts (here, names of endangered mammals) but also the language and the style used by others to describe them.

< top >

< num > Number: 304

< title > Endangered Species (Mammals)

< desc > Description:

Compile a list of mammals that are considered to be endangered, identify their habitat and, if possible, specify what threatens them.

< narr > Narrative:

Any document identifying a mammal as endangered is relevant. Statements of authorities disputing the endangered status would also be relevant. A document containing information on habitat and populations of a mammal identified elsewhere as endangered would also be relevant even if the document at hand did not identify the species as endangered. Generalized statements about endangered species without reference to specific mammals would not be relevant.

< expd > Expanded:

---

[2]TREC-6 database consisted of approx. 2 GBytes of documents from Associated Press newswire, Wall Street Journal, Financial Times, Federal Register, FBIS and other sources (Harman & Voorhees 1998).

The Service is responsible for eight species of marine mammals under the jurisdiction of the Department of the Interior, as assigned by the Marine Mammal Protection Act of 1972. These species are polar bear, sea and marine otters, walrus, manatees (three species) and dugong. The report reviews the Service's marine mammal-related activities during the report period.

The U.S. Fish and Wildlife Service had classified the primate as a "threatened" species, but officials said that more protection was needed in view of recent studies documenting a drastic decline in the populations of wild chimps in Africa.

The Endangered Species Act was passed in 1973 and has been used to provide protection to the bald eagle and grizzly bear, among other animals.

Under the law, a designation of a threatened species means it is likely to become extinct without protection, whereas extinction is viewed as a certainty for an endangered species.

The bear on California's state flag should remind us of what we have done to some of our species. It is a grizzly. And it is extinct in California and in most other states where it once roamed.

< /top >

In the next section we describe the summarization process in detail.

## Robust text summarization

Perhaps the most difficult problem in designing an automatic text summarization is to define what a summary is, and how to tell a summary from a non-summary, or a good summary from a bad one. The answer depends in part upon who the summary is intended for, and in part upon what it is meant to achieve, which in large measure precludes any objective evaluation. For most of us, a summary is a brief synopsis of the content of a larger document, an abstract recounting the main points while suppressing most details. One purpose of having a summary is to quickly learn some facts, and decide what you want to do with the entire story. Therefore, one important evaluation criterion is the tradeoff between the degree of compression afforded by the summary, which may result in a decreased accuracy of information, and the time required to review that information. This interpretations is particularly useful, though it isn't the only one acceptable, in summarizing news and other report-like documents. It is also well suited for evaluating the usefulness of summarization in context of an information retrieval system, where the user needs to rapidly and efficiently review the documents returned from search for an indication of relevance and, possibly, to see which aspect of relevance is present.

Our early inspiration, and a benchmark, have been the Quick Read Summaries, posted daily off the front page of New York Times on-line edition (http://www.nytimes.com). These summaries, produced manually by NYT staff, are assembled out of

1260

passages, sentences, and sometimes sentence fragments taken from the main article with very few, if any, editorial adjustments. The effect is a collection of perfectly coherent tidbits of news: the who, the what, and when, but perhaps not why. This kind of summarization, where appropriate passages are extracted from the original text, is very efficient, and arguably effective, because it doesn't require generation of any new text, and thus lowers the risk of misinterpretation. It is also relatively easier to automate, because we only need to identify the suitable passages among the other text, a task that can be accomplished via shallow NLP and statistical techniques.[3]

It has been noted, eg., (Rino & Scott 1994), (Weissberg & Buker 1990), that certain types of texts, such as news articles, technical reports, research papers, etc., conform to a set of style and organization constraints, called the Discourse Macro Structure (DMS) which help the author to achieve a desired communication effect. News reports, for example, tend to be built hierarchically out of components which fall roughly into one of the two categories: the what's-the-news category, and the optional background category. The background, if present, supplies the context necessary to understand the central story, or to make a follow up story self-contained. This organization is often reflected in the summary, as illustrated in the example below from NYT 10/15/97, where the highlighted portion provides the background for the main news:

Spies Just Wouldn't Come In From Cold War, Files Show

*Terry Squillacote was a Pentagon lawyer who hated her job. Kurt Stand was a union leader with an aging beatnik's slouch. Jim Clark was a lonely private investigator.* [A 200-page affidavit filed last week by] the Federal Bureau of Investigation says the three were out-of-work spies for East Germany. And after that state withered away, it says, they desperately reached out for anyone who might want them as secret agents.

In this example, the two passages are non-consecutive paragraphs in the original text; the string in the square brackets at the opening of the second passage has been omitted in the summary. Here the human summarizer's actions appear relatively straightforward, and it would not be difficult to propose an algorithmic method to do the same. This may go as follows:

1. Choose a DMS template for the summary; e.g., Background+News.

---

[3]This approach is contrasted with a far more difficult method of summarizing text "in your own words." Computational attempts at such discourse-level and knowledge-level summarization include (Ono, Sumita & Miike 1994), (McKeown & Radev 1995), (DeJong 1982), and (Lehnert 1981).

2. Select appropriate passages from the original text and fill the DMS template.

3. Assemble the summary in the desired order; delete extraneous words.

We have used this method to build our automated summarizer. We overcome the shortcomings of sentence-based summarization by working on paragraph level instead.[4] The summarizer has been applied to a variety of documents, including Associated Press newswires, articles from the New York Times, Wall Street Journal, Financial Times, San Jose Mercury, as well as documents from the Federal Register, and Congressional Record. The program is domain independent, and it can be easily adapted to most European languages. It is also very robust: we used it to derive summaries of thousands of documents returned by an information retrieval system. It can work in two modes: generic and topical. In the generic mode, it captures the main topic of a document; in the topical mode, it takes a user supplied statement of interest and derives a summary related to this topic. The topical summary is usually different than the generic summary of the same document.

## Deriving automatic summaries

Each component of a summary DMS needs to be instantiated by one or more passages extracted from the original text. Initially, all eligible passages (i.e., explicitly delineated paragraphs) within a document are potential candidates for the summary. As we move through text, paragraphs are scored for their summary-worthiness. The final score for each passage, normalized for its length, is a weighted sum of a number of minor scores, using the following formula:[5]

$$score(paragraph) = \frac{1}{l} \bullet \sum_h w_h \bullet S_h \qquad (1)$$

where $S_h$ is a minor score calculated using metric $h$; $w_h$ is the weight reflecting how effective this metric is in general; $l$ is the length of the segment.

The following metrics are used to score passages considered for the main news section of the summary DMS. We list here only the criteria which are the

---

[4]Refer to (Luhn 1958) (Paice 1990) (Rau, Brandow & Mitze 1994) (Kupiec, Pedersen & Chen 1995) for sentence-based summarization approaches.

[5]The weights $w_h$ are trainable in a supervised mode, given a corpus of texts and their summaries, or in an unsupervised mode as described in (Strzalkowski & Wang 1996). For the purpose of the experiments described here, these weights have been set manually.

most relevant for generating summaries in context of an information retrieval system.

1. Words and phrases frequently occurring in a text are likely to be indicative of its content, especially if such words or phrases do not occur often elsewhere in the database. A weighted frequency score, similar to $tf^*idf$ used in automatic text indexing is applicable. Here, $idf$ stands for the inverted document frequency of a term.

2. Title of a text is often strongly related to its content. Therefore, words and phrases from the title repeated in text are considered as important indicators of content concentration within a document.

3. Noun phrases occurring in the opening sentences of multiple paragraphs tend to be indicative of the content. These phrases, along with words from the title receive premium scores.

4. In addition, all significant terms in a passage (i.e., other than the common stopwords) are ranked by a passage-level inverted frequency distribution, e.g., $N/pf$, where $pf$ is the number of passages containing the term and $N$ is the total number of passages contained in a document.

5. For generic-type summaries, in case of score ties the passages closer to the beginning of a text are preferred to those located towards the end.

The process of passage selection as described here resembles query-based document retrieval. The "documents" here are the passages, and the "query" is a set of words and phrases found in the document's title and in the openings of some paragraphs. Note that the summarizer scores both single- and multi-paragraph passages, which makes it more independent from any particular physical paragraph structure of a document.

## Supplying the background passage

The background section supplies information that makes the summary self-contained. For example, a passage selected from a document may have significant links, both explicit and implicit, to the surrounding context, which if severed are likely to render the passage uncomprehensible, or even misleading. The following passage illustrates the point:

"Once again this demonstrates the substantial influence Iran holds over terrorist kidnapers," Redman said, adding that it is not yet clear what prompted Iran to take the action it did.

Adding a background paragraph makes this a far more informative summary:

Both the French and Iranian governments acknowledged the Iranian role in the release of the three French hostages, Jean-Paul Kauffmann, Marcel Carton and Marcel Fontaine.

"Once again this demonstrates the substantial influence Iran holds over terrorist kidnapers," Redman said, adding that it is not yet clear what prompted Iran to take the action it did.

Below are three main criteria we consider to decide if a background passage is required, and if so, how to get one.

1. One indication that a background information may be needed is the presence of outgoing references, such as anaphors. If an anaphor is detected within the first N (=6) items (words, phrases) of the selected passage, the preceding passage is appended to the summary. Anaphors and other references are identified by the presence of pronouns, definite noun phrases, and quoted expressions.

2. Initially the passages are formed from single physical paragraphs, but for some texts the required information may be spread over multiple paragraphs so that no clear "winner" can be selected. Subsequently, multi-paragraph passages are scored, starting with pairs of adjacent paragraphs.

3. If the selected main summary passage is shorter than L characters, then the passage following it is added to the to the summary. The value of L depends upon the average length of the documents being summarized, and it was set as 100 characters for AP newswire articles. This helps avoiding choppy summaries from texts with a weak paragraph structure.

## Implementation and evaluation

The summarizer has been implemented as a demonstration system, primarily for news summarization. In general we are quite pleased with the system's performance. The summarizer is domain independent, and can effectively process a range of types of documents. The summaries are quite informative with excellent readability. They are also quite short, generally only 5 to 10% of the original text and can be read and understood very quickly.

As discussed before, we have included the summarizer as a helper application within the user interface to the natural language information retrieval system. In this application, the summarizer is used to derive query-related summaries of documents returned from database search. The summarization method used here is the same as for generic summaries described thus far, with the following exceptions:

1262

1. The passage-search "query" is derived from the user's document search query rather than from the document title.

2. The distance of a passage from the beginning of the document is not considered towards its summary-worthiness.

The topical summaries are read by the users to quickly decide their relevance to the search topic and, if desired, to expand the initial information search statement in order to produce a significantly more effective query. The following example shows a topical (query-guided summary) and compares it to the generic summary (we abbreviate SGML for brevity).

INITIAL SEARCH STATEMENT:

< title > Evidence of Iranian support for Lebanese hostage takers.

< desc > Document will give data linking Iran to groups in Lebanon which seize and hold Western hostages.

FIRST RETRIEVED DOCUMENT (TITLE):

Arab Hijackers' Demands Similar To Those of Hostage-Takers in Lebanon

SUMMARIZER TOPIC:

Evidence of Iranian support for Lebanese hostage takers

TOPICAL SUMMARY (used for expansion):

Mugniyeh, 36, is a key figure in the security apparatus of Hezbollah, or Party of God, an Iranian-backed Shiite movement believed to be the umbrella for factions holding most of the 22 foreign hostages in Lebanon.

GENERIC SUMMARY (for comparison):

The demand made by hijackers of a Kuwaiti jet is the same as that made by Moslems holding Americans hostage in Lebanon - freedom for 17 pro-Iranian extremists jailed in Kuwait for bombing U.S. and French embassies there in 1983.

PARTIALLY EXPANDED SEARCH STATEMENT:

< title > Evidence of Iranian support for Lebanese hostage takers.

< desc > Document will give data linking Iran to groups in Lebanon which seize and hold Western hostages.

< expd > Mugniyeh, 36, is a key figure in the security apparatus of Hezbollah, or Party of God, an Iranian-backed Shiite movement believed to be the umbrella for factions holding most of the 22 foreign hostages in Lebanon.

## Overview of the NLIR System

The Natural Language Information Retrieval System (NLIR)[6] as been designed as a series of parallel text processing and indexing "streams". Each stream constitutes an alternative representation of the database obtained using different combination of natural language processing steps. The purpose of NL processing is to obtain a more accurate content representation than that based on words alone, which will in turn lead to improved performance. The following term extraction steps correspond to some of the streams used in our system:

---

[6]For more details, see (Strzalkowski 1995), (Strzalkowski et al. 1997)

1. Elimination of stopwords: Documents are indexed using original words minus selected "stopwords" that include all closed-class words (determiners, prepositions, etc.)

2. Morphological stemming: Words are normalized across morphological variants using a lexicon-based stemmer.

3. Phrase extraction: Shallow text processing techniques, including part-of-speech tagging, phrase boundary detection, and word co-occurrence metrics are used to identify relatively stable groups of words, e.g., *joint venture*.

4. Phrase normalization: Documents are processed with a syntactic parser, and "Head+Modifier" pairs are extracted in order to normalize across syntactic variants and reduce to a common "concept", e.g., *weapon+proliferate*.

5. Proper name extraction: Names of people, locations, organizations, etc. are identified.

Search queries, after appropriate processing, are run against each stream, i.e., a phrase query against the phrase stream, a name query against the name stream, etc. The results are obtained by merging ranked lists of documents obtained from searching all streams. This allows for an easy combination of alternative retrieval methods, creating a meta-search strategy which maximizes the contribution of each stream. Different information retrieval systems can used as indexing and search engines each stream. In the experiments described here we used Cornell's SMART (version 11) (Buckley, et al. 1995).

## TREC Evaluation Results

Table 1 lists selected runs performed with the NLIR system on TREC-6 database using 50 queries (TREC topics) numbered 301 through 350. The expanded query runs are contrasted with runs obtained using TREC original topics using NLIR as well as Cornell's SMART (version 11) which serves here as a benchmark. The first two columns are automatic runs, which means that there was no human intervention in the process at any time. Since query expansion requires human decision on summary selection, these runs (columns 3 and 4) are classified as "manual", although most of the process is automatic. As can be seen, query expansion produces an impressive improvement in precision at all levels. Recall figures are shown at 1000 retrieved documents.

Query expansion appears to produce consistently high gains not only for different sets of queries but

Table 1: Performance improvement for expanded queries

| queries: | original | original | expanded | expanded |
|----------|----------|----------|----------|----------|
| SYSTEM | SMART | NLIR | SMART | NLIR |
| PRECISION | | | | |
| Average | 0.1429 | 0.1837 | 0.2672 | 0.2859 |
| %change | | +28.5 | +87.0 | +100.0 |
| At 10 docs | 0.3000 | 0.3840 | 0.5060 | 0.5200 |
| %change | | +28.0 | +68.6 | +73.3 |
| At 30 docs | 0.2387 | 0.2747 | 0.3887 | 0.3940 |
| %change | | +15.0 | +62.8 | +65.0 |
| At 100 doc | 0.1600 | 0.1736 | 0.2480 | 0.2574 |
| %change | | +8.5 | +55.0 | +60.8 |
| Recall | 0.57 | 0.53 | 0.61 | 0.62 |
| %change | | -7.0 | +7.0 | +8.7 |

also for different systems: we asked other groups participating in TREC to run search using our expanded queries, and they reported similarly large improvements.

Finally, we may note that NLP-based indexing has also a positive effect on overall performance, but the improvements are relatively modest, particularly on the expanded queries. A similar effect of reduced effectiveness of linguistic indexing has been reported also in connection with improved term weighting techniques.

## Conclusions

We have developed a method to derive quick-read summaries from news-like texts using a number of shallow NLP and simple quantitative techniques. The summary is assembled out of passages extracted from the original text, based on a pre-determined DMS template. This approach has produced a very efficient and robust summarizer for news-like texts.

We used the summarizer, via the QET interface, to build effective search queries for an information retrieval system. This has been demonstrated to produce dramatic performance improvements in TREC evaluations. We believe that this query expansion approach will also prove useful in searching very large databases where obtaining a full index may be impractical or impossible, and accurate sampling will become critical.

## References

Buckley, Chris, Amit Singhal, Mandar Mitra, Gerard Salton. 1995. "New Retrieval Approaches Using SMART: TREC 4". Proceedings of TREC-4 Conference, NIST Special Publication 500-236.

DeJong, G.G., 1992. An overview of the FRUMP system, Lehnert, W.G. and M.H. Ringle (eds), Strategies for NLP, Lawrence Erlbaum, Hillsdale, NJ.

Harman, Donna. 1988. "Towards interactive query expansion." Proceedings of ACM SIGIR-88, pp. 321-331.

Harman, Donna, and Ellen Voorhees (eds). 1998. The Text Retrieval Conference (TREC-6). NIST Special Publication (to appear).

Kupiec,J., J. Pedersen and F. Chen, 1995. A trainable document summarizer, Proceedings of ACM SIGIR-95, pp. 68-73.

Lehnert, W.G., 1981. Plots Units and Narrative summarization, Cognitive Science, 4, pp 293-331.

Luhn, H.P., 1958. The automatic creation of literature abstracts, IBM Journal, Apr, pp. 159-165.

McKeown, K.R. and D.R. Radev, 1995. Generating Summaries of Multiple News Articles, Proceedings of ACM SIGIR-95

Proceedings of 5th Message Understanding Conference, San Francisco, CA:Morgan Kaufman Publishers. 1993.

Ono, K., K. Sumita and S.Miike, 1994. Abstract Generation based on Rhetorical Structure Extraction, COLING94, vol 1, pp 344-348, Kyoto, Japan.

Paice, C.D., 1990. Constructing literature abstracts by computer: techniques and prospects, Information Processing and Management, vol 26 (1), pp 171-186.

Rau, L.F., R. Brandow and K. Mitze, 1994. Domain-independent summarization of news, Summarizing text for intelligent communication, page 71-75, Dagstuhl, Gemany.

Rino, L.H.M. and D. Scott, 1994. Content selection in summary generation, Third International Conference on the Cognitive Science of NLP, Dublin City University, Ireland.

Rocchio, J. J. 1971. "Relevance Feedback in Informatio Retrieval." In Salton, G. (Ed.), The SMART Retrieval System, pp. 313-323. Prentice Hall, Inc., Englewood Cliffs, NJ.

Strzalkowski, Tomek, Jin Wang, and Bowden Wise. 1998. "A Robust Practical Text Summarization." Proceedings of AAAI Spring Symposium on Intelligent Text Summarization (to appear).

Strzalkowski, Tomek, Fang Lin, Jose Perez-Carballo, and Jin Wang. 1997. "Natural Language Information Retrieval: TREC-6 Report." Proceedings of TREC-6 conference.

Strzalkowski, Tomek, Louise Guthrie, Jussi Karlgren, Jim Leistensnider, Fang Lin, Jose Perez-Carballo, Troy Straszheim, Jin Wang, and Jon Wilding. 1997. "Natural Language Information Retrieval: TREC-5 Report." Proceedings of TREC-5 conference.

Strzalkowski, Tomek. 1995. "Natural Language Information Retrieval" Information Processing and Management, Vol. 31, No. 3, pp. 397-417. Pergamon/Elsevier.

Strzalkowski, Tomek. and Jin Wang, 1996. A Self-Learning Universal Concept Spotter, Proceedings of COLING-96, pp. 931-936.

Tipster Text Phase 2: 24 month Conference, Morgan-Kaufmann. 1996

Voorhees, Ellen M. 1994. "Query Expansion Using Lexical-Semantic Relations." Proceedings of ACM SIGIR'94, pp. 61-70.

Weissberg, R. and S. Buker, 1990. Writing up Research: Experimental Research Report Writing for Student of English, Prentice Hall, Inc.