Concept Identification and Presentation in the Context of Technical Text Summarization

Horacio Saggion*and Guy Lapalme
Département d'Informatique et Recherche Opérationnelle
Université de Montréal
CP 6128, Succ Centre-Ville
Montréal, Québec, Canada, H3C 3J7
Fax: +1-514-343-5834
{saggion,lapalme}@iro.umontreal.ca

Abstract

We describe a method of text summarization that produces indicative-informative abstracts for technical papers. The abstracts are generated by a process of conceptual identification, topic extraction and re-generation. We have carried out an evaluation to assess indicativeness and text acceptability relying on human judgment. The results so far indicate good performance in both tasks when compared with other summarization technologies.

1 Introduction

We have specified a method of text summarization which produces indicative-informative abstracts for technical documents. The method was designed to identify the "topics" of a document and present them in an indicative abstract. Eventually, they can be elaborated in specific ways.

In Figure 1, we present an indicative abstract for the document "Facilitating designer-customer communication in the World Wide Web" (Internet Research: Electronic Networking Applications and Policy, Vol 8, Issue 5,1998) produced with our implementation of this method. The abstract includes a list of topics which are terms appearing in the automatic abstract (e.g. **WebShaman**) or obtained from the source document by the process of term expansion (e.g. **WWW technique** obtained from **technique**). It also includes term elaborations which can be used to answer specific questions about the topics such as **what is topic? how topic is used? who developed topic?** and **what are the advantages of topic?**

In this paper, we will describe how we dealt with the problem of content selection and presentation and how we have evaluated our method of text summarization.

2 Text Summarization

The process of producing a summary from a source text consists of the following steps: (i) the interpretation of the text; (ii) the extraction of the relevant information which ideally includes the "topics" of the source; (iii) the condensation of the extracted information and construction of a summary representation; and (iv) the presentation of the summary representation to the reader in natural language.

While some techniques exist for producing summaries for domain independent texts (Luhn, 1958; Marcu, 1997) it seems that domain specific texts require domain specific techniques (DeJong, 1982; Paice and Jones, 1993). In our case, we are dealing with technical articles which are the result of the complex process of scientific inquiry that starts with the identification of a knowledge problem and eventually culminates with the discovery of an answer to it. Even if authors of technical articles write about several concepts in their articles, not all of them are topics. In order to address the issue of topic identification, content selection and presentation, we have studied alignments (manually produced) of sentences from professional abstracts with sentences from

Abstract Introducing the Topics

Virtual prototyping is a technique which has been suggested for use in, for example, telecommunication product development as a high-end technology to achieve a quick digital model that could be used in the same way as a real prototype. Presents the design rationale of WebShaman, starting from the concept design perspective by introducing a set of requirements to support communication via a concept model between industrial designer and a customer. In the article, the authors suggest that virtual prototyping in collaborative use between designers is a potential technique to facilitate design and alleviate the problems created by geographical distance and complexities in the work between different parties. The technique, was implemented in the VRP project, allows component level manipulation of a virtual prototype in a WWW (World Wide Web) browser. The user services, the software architecture, and the techniques of WebShaman were developed iteratively during the fieldwork in order to illustrate the ideas and the feasibility of the system. The server is not much different from the other servers constructed to support synchronous collaboration.

Identified Topics: **3D model** - **VIRPI project** - **WWW** - **WWW technique** - **WebShaman** - **CAD system** - **conceptual model** - **customer** - **object-oriented model** - **product** - **product concept** - **product design** - **requirement** - **simulation model** - **smart virtual prototype** - **software component** - **system** - **technique** - **technology** - **use** - **virtual component** - **virtual prototype** - **virtual prototype system** - **virtual prototyping**

Information about the Topics

An example of **a conceptual model, a pen-shaped wireless user interface** for a mobile telephone.

A virtual prototype is a computer-based simulation of a prototype or a subsystem with a degree of functional realism, comparable to that of a physical prototype.

A computer system implementing **the high-end aspects of virtual prototyping has been developed** in the VRP project (VRP, 1998) at VTT Electronics, in Oulu, Finland.

The two-and-a-half-year **VIRPI project consists of three parts**.

Nowadays, **CAD (computer-aided design) systems are used** as an aid in industrial, mechanical and electronics design for the specification and **development of a product**.

A virtual prototype system can be used for concept testing in the early phase of product development.

Figure 1: Indicative Abstract, Topics and Topic Elaboration

source documents. One of the alignments is presented in Table 1. The first column contains the information of the professional abstract. The second and third columns contain the information from the source document that matches the sentences of the professional abstract, and its location in the source document. We have produced 100 of these tables containing a total of 309 sentences of professional abstracts aligned with 568 sentences of source documents.

These alignments allowed us to identify on one hand, concepts, relations and types of information usually conveyed in abstracts; and on the other hand, valid transformations in the source in order to produce a compact and coherent text. The transformations include *verb transformation, concept deletion, concept reformulation, structural deletion, parenthetical deletion, clause deletion, acronym expansion,*

Professional Abstract	Source Document	P/T
Presents a more efficient Distributed Breadth-First Search algorithm for an asynchronous communication network.	Efficient distributed breadth-first search algorithm.	-/Title
	In *this paper we have presented* a more efficient distributed algorithm which construct a breadth-first search tree in an asynchronous communication network	Lst/-
Presents a model and *gives* an overview of related research.	First *we present* a model and *give* overview of related research.	1st/-
Analyzes the complexity of the algorithm, and *gives* some examples of performance on typical networks.	*We analyse* the complexity of our algorithm, and *give* some examples of performance on typical networks.	1st/-

Table 1: LISA Abstract 1955 - Source Document: "Efficient distributed breadth-first search algorithm." S.A.M. Makki. Computer Communications, 19(8) Jul 96, p628-36.

abbreviation, merge and *split*. In our corpus, 89% of the sentences from the professional abstracts included at least one transformation. Results of the corpus study are detailed in (Saggion and Lapalme, 1998) and (Saggion and Lapalme, 2000).

We have identified a total of 52 different types of information (coming from the corpus and from technical articles) for technical text summarization that we use to identify some of the main themes. These types include: **the explicit topic** of the document, **the situation**, **the identification of the problem**, **the identification of the solution**, **the research goal**, **the explicit topic** of a section, **the authors' development**, **the inferences**, **the description** of a topical entity, **the definition** of a topical entity, **the relevance** of a topical entity, **the advantages**, etc. Information types are classified as indicative or informative depending on the type of abstract they contribute to (i.e. **the topic** of a document is indicative while **the description** of a topical entity is informative). Types of information are identified in sentences of the source document using co-occurrence of concepts and relations and specific linguistic patterns. Technical articles from different domains refer to specific concepts and relations (diseases and treatments in Medicine, atoms and chemical reactions in Chemistry, and theorems and proofs in Mathematics). We have focused on concepts and relations that are common across domains such as `problem`, `solution`, `research need`, `experiment`, `relevance`, `researchers`, etc.

3 Text Interpretation

Our approach to text summarization is based on a superficial analysis of the source document and on the implementation of some text re-generation techniques such as merging of topical information, re-expression of concepts and acronym expansion. The article (plain text in English without mark-up) is segmented in main units (title, author information, author abstract, keywords, main sections and references) using typographic information and some keywords. Each unit is passed through a bi-pos statistical tagger. In each unit, the system identifies titles, sentences and paragraphs, and then, sentences are interpreted using finite state transducers identifying and packing linguistic constructions and domain specific constructions. Following that, a conceptual dictionary that relates lexical items to domain concepts and relations is used to associate semantic tags to the different structural elements in the sentence. Subsequently, terms (canonical form of noun groups), their associated semantic (head of the noun group) and theirs positions are extracted from each sentence and stored in an AVL tree (`term tree`) along with their frequency. A `conceptual index` is created which specifies to which particular type of information each sentence could contribute. Finally, terms and words are extracted from titles and stored in a list (the `topical structure`) and acronyms and their expansions are recorded.

3.1 Content Selection

In order to represent types of information we use templates. In Table 2, we present the **Topic of the Document**, **Topic of the Section** and **Signaling Information** templates. Also presented are some indicative and informative patterns. Indicative patterns contain variables, syntactic constructions, domain concepts and relations. Informative patterns also include one specific position for the topic under consideration. Each element of the pattern matches one or more elements of the sentence (conceptual, syntactic and lexical elements match one element while variables match zero or more).

3.1.1 Indicativeness

The system considers sentences that were identified as carrying indicative information (their position is found in the `conceptual index`). Given a sentence S and a type of information T the system verifies if the sentence matches some of the patterns associated with type T. For each matched pattern, the system extracts information from the sentence and instantiates a template of type T. For example, the `Content` slot of the **problem identification** template is instantiated with all the sentence (avoiding references, structural elements and parenthetical expressions) while the `What` slot of the **topic of the document** template is instantiated with a parsed sentence fragment to the left or to the right of the `make known` relation depending on the attribute voice of the verb (active vs. passive). All the instantiated templates constitute the Indicative Data Base (IDB).

The system matches the `topical structure` with the `topic candidate` slots from the IDB. The system selects one template for each term in that structure: the one with the greatest `weight` (heuristics are applied if there are more than one). The selected templates constitute the `indicative content` and the terms appearing in the `topic candidate` slots and their expansions constitute the `potential topics` of the document. Expansions are obtained looking for terms in the `term tree` sharing the semantic of some terms in the `indicative content`.

The `indicative content` is sorted using positional information and the following conceptual order: **situation**, **need for research**, **problem**, **solution**, **entity introduction**, **topical information**, **goal of conceptual entity**, **focus of conceptual entity**, **methodological aspects**, **inferences** and **structural information**. Templates of the same type are grouped together if they appeared in sequence in the list. The types considered in this process are: **the topic**, **section topic** and **structural information**. The sorted templates constitute the `text plan`.

3.1.2 Informativeness

For each `potential topic` and sentence where it appears (that information is found on the `term tree`) the system verifies if the sentence contains an informative marker (`conceptual index`) and satisfies an informative pattern. If so, the `potential topic` is considered a `topic` of the document and a link will be created between the topic and the sentence which will be part of the informative abstract.

4 Content Presentation

Our approach to text generation is based on the regularities observed in the corpus of professional abstracts and so, it does not implement a general theory of text generation by computers. Each element in the `text plan` is used to produce a sentence. The structure of the sentence depends on the type of template. The information about the **situation**, the **problem**, the **need for research**, etc. is reported as in the original document with few modifications (concept re-expression). Instead other types require additional re-generation: for the **topic of the document** template the generation procedure is as follows: (i) the verb form for the predicate in the `Predicate` slot is generated in the `present tense` (topical information is always reported in present tense), `3rd person of singular` in `active voice` at the `beginning of the sentence`; (ii) the parsed sentence fragment from the `What` slot is generated in the middle of the sentence (so the appropriate case for the first element has to be generated); and (iii) a full stop is generated. This schema of generation avoids

Topic of the Document	
Type:	topic
Id:	integer identifier
Predicate:	instance of `make known`
Where:	instance of {`research paper, study, work, research`}
Who:	instance of {`research paper, author, study, work, research, none`}
What:	parsed sentence fragment
Position:	section and sentence id
Topic candidates:	list of terms from the `What` filler
Weight:	number
Topic of Section	
Type:	sec_desc
Id:	integer identifier
Predicate:	instance of `make known`
Section:	instance of `paper component`
Argument:	parsed sentence fragment
Position:	section and sentence id
Topic candidates:	list of terms from the `Argument` filler
Weight:	number
Signaling Information	
Type:	structure_2
Id:	integer identifier
Predicate:	instance of `show graphical material`
Structural:	instance of `structural element`
Argument:	parsed sentence fragment
Position:	section and sentence id
Topic candidates:	list of terms from the `Argument` filler
Weight:	number

Signaling (indicative)	$SKIP_1$ + `structural` + $SKIP_2$ + `show graphically` + $ARGUMENT$ + `eos`
Topic (indicative)	`noun group` + `author` + `make known` + `preposition` + `research paper` + $DESCRIPTION$ + `eos`
Author's Goal (indicative)	$SKIP_1$ + `goal of author` + `define` + $GOAL$ + `eos`
Goal of *TOPIC* (informative)	$SKIP$ + `goal` + `preposition` + $TOPIC$ + `define` + $GOAL$ + `eos`
Definition of *TOPIC* (informative)	$SKIP$ + $TOPIC$ + `define` + `noun group`

Table 2: Templates and Patterns.

the formulation of expressions like "X will be presented", "X have been presented" or "We have presented here X" which are usually found on source documents but which are awkward in the context of the abstract text-type. Note that each type of information prescribes its own schema of generation.

Some elements in the parsed sentence frag-

ment require re-expression while others are presented in "the words of the author." If the system detects an acronym without expansion in the string it would expand it and record that situation in order to avoid repetitions. Note that as the templates contain parsed sentence fragments, the correct punctuation has to be re-generated. For merged templates the generator implements the following patterns

of production: if n adjacent templates are to be presented using the same predicate, only one verb will be generated whose argument is the conjunction of the arguments from the n templates. If the sequence of templates have no common predicate, the information will be presented as a conjunction of propositions. These patterns of sentence production are exemplified in Table 3.

The elaboration of the topics is presented upon reader's demand. The information is presented in the order of the original text. The informative abstract is the information obtained by this process as it is shown in Figure 1.

5   Limitations of the Approach

Our approach is based on the empirical examination of abstracts published by second services. In our first study, we examined 100 abstracts and source documents in order to deduce a conceptual and linguistic model for the task of summarization of technical articles. Then, we expanded the corpus with 100 more items in order to validate the model. We believe that the concepts, relations and types of information identified account for interesting phenomena appearing in the corpus and constitute a sound basis for text summarization. Nevertheless, we have identified only a few linguistic expressions used in order to express particular elements of the conceptual model (241 domain verbs, 163 domain nouns, 129 adjectives, 174 indicative patterns, 87 informative patterns). This is because we are mainly concerned with the development of a general method of automatic abstracting and the task of constructing such linguistic resources is time consuming as recent work have shown (Minel et al., 2000).

The implementation of our method relies on state-of-the-art techniques in natural language processing including noun and verb group identification and conceptual tagging. The interpreter relies on the output produced by a shallow text segmenter and on a statistical POS-tagger. Our prototype only analyses sentences for the specific purpose of text summarization and implements some patterns of generation observed in the corpus. Additional analysis could

be done on the obtained representation to produce better results.

6   Related Work

(Paice and Jones, 1993) have already addressed the issue of content identification and expression in technical summarization using templates, but while they produced indicative abstracts for a specific domain, we are producing domain independent indicative-informative abstracts. Being designed for one specific domain, their abstracts are fixed in structure while our abstracts are dynamically constructed. Radev and McKeown (1998) also used instantiated templates, but in order to produce summaries of multiple documents in one specific domain. They focus on the generation of the text while we are addressing the overall process of automatic abstracting. Our concern regarding the presentation of the information is now being addressed by other researchers as well (Jing and McKeown, 1999).

7   Evaluating Content and Quality in Text Summarization

Abstracts are texts used in tasks such as assessing the content of the document and deciding if the source is worth reading. If text summarization systems are designed to fulfill those requirements, the generated texts have to be evaluated according to their intended function and its quality. The quality and success of human produced abstracts have already been addressed in the literature (Grant, 1992; Gibson, 1993) using linguistic criteria such as cohesion and coherence, thematic structure, sentence structure and lexical density. But in automatic text summarization, this is an emergent research topic. (Minel et al., 1997) have proposed two methods of evaluation addressing the content of the abstract and its quality. For content evaluation, they asked human judges to classify summaries in broad categories and also verify if the key ideas of source documents are appropriately expressed in the summaries. For text quality, they asked human judges to identify problems such as dangling anaphora and broken textual segments and also to make subjective judgments about readability. In the context of the TIPSTER program, (Firmin and Chrzanowski, 1999) and (Mani et al., 1998) also used a categorization task using TREC topics. For text

Re-Generated Sentences	Sentences from Source Documents
Illustrates the principle of virtual prototyping and the different techniques and models required.	Figure 1 Virtual prototyping models and techniques illustrates the principle of virtual prototyping and the different techniques and models required.
Presents the mechanical and electronic design of the robot harvester including all subsystems, namely, fruit localisation module, harvesting arm and gripper-cutter as well as the integration of subsystems and the specific mechanical design of the picking arm addressing the reduction of undesirable dynamic effects during high velocity operation.	After a brief introduction, we present the mechanical and electronic design of the robot harvester including all subsystems, namely , fruit localisation module, harvesting arm and gripper-cutter as well as the integration of subsystems.
	Throughout this work, we present the specific mechanical design of the picking arm addressing the reduction of undesirable dynamic effects during high velocity operation.
Shows configuration of the robotic fruit harvester Agribot and schematic view of the detaching tool.	The final prototype consists of two jointed harvesting arms mounted on a human guided vehicle as shown schematically in Figure 1 Configuration of the robotic fruit harvester Agribot.
	Schematic representation of the operations involved in the detaching step can be seen in Figure 5 Schematic view of the detaching tool and operation.
PAWS (the programmable automated welding system) was designed to provide an automated means of planning, controlling, and performing critical welding operations for improving productivity and quality.	PAWS was designed to provide an automated means of planning, controlling, and performing critical welding operations for improving productivity and quality.
Describes HuDL (local autonomy) in greater detail; discusses system integration and the IMA (the intelligent machine architecture); and also gives an example implementation.	Section 2 describes HuDL in greater detail and section 3 discusses system integration and the IMA .
	An example implementation is given in section 4 and section 5 contains the conclusions.

Table 3: Re-Generated Sentences

quality, they addressed subjective aspects such as the length of the summary, its intelligibility and its usefulness. We have carried out an evaluation of our summarization method in order to assess the function of the abstract and its text quality.

7.1 Experiment

We compared abstracts produced by our method with abstracts produced by Microsoft'97 Summarizer and with others published with source documents (usually author abstracts). We have chosen Microsoft'97 Summarizer because, even if it only produces extracts, it was the only summarizer available in order to carry out this evaluation and because it has already been used in other evaluations (Marcu, 1997; Barzilay and Elhadad, 1997).

In order to evaluate content, we presented judges with randomly selected abstracts and five lists of keywords (content indicators). The judges had to decide to which list of keywords the abstract belongs given that different lists share some keywords and that they belong to the same technical domain. Those lists were obtained from the journals where the source documents were published. The idea behind this evaluation is to see if the abstract convey the very essential content of the source document.

In order to evaluate the quality of the text, we asked the judges to provide an acceptability score between 0-5 for the abstract (0 for unacceptable and 5 for acceptable) based on the following criteria taken from (Rowley, 1982) (they were only suggestions to the evaluators and were not enforced): good spelling and grammar; clear indication of the topic of the source document; impersonal style; one paragraph; conciseness; readable and under-

standable; acronyms are presented along with their expansions; and other criteria that the judge considered important as an experienced reader of abstracts of technical documents.

We told the judges that we would consider the abstracts with scores above 2.5 as acceptable. Some criteria are more important than other, for example judges do not care about impersonal style but care about readability.

7.1.1 Materials
Source Documents: we used twelve source documents from the journal Industrial Robots found on the Emerald Electronic Library (all technical articles). The articles were downloaded in plain text format. These documents are quite long texts with an average of 23K characters (minimum of 11K characters and a maximum of 41K characters). They contain an average of 3472 words (minimum of 1756 words and a maximum of 6196 words excluding punctuation), and an average of 154 sentences (with a minimum of 85 and a maximum of 288).

Abstracts: we produced twelve abstracts using our method and computed the compression ratio in number of words, then we produced twelve abstracts by Microsoft'97 Summarizer[1] using a compression rate at least as high as our (i.e. if our method produced an abstract with a compression rate of 3.3% of the source, we produced the Microsoft abstract with a compression rate of 4% of the source). We extracted the twelve abstracts and the twelve lists of keywords published with the source documents. We thus obtained 36 different abstracts and twelve lists of keywords.

Forms: we produced 6 different forms each containing six different abstracts randomly[2] chosen out of twelve different documents (for a total of 36 abstracts). Each abstract was printed in a different page. It included 5 lists of keywords, a field to be completed with the quality score as-

---

[1]We had to format the source document in order for the Microsoft Summarizer to be able to recognize the structure of the document (titles, sections, paragraphs and sentences).

[2]Random numbers for this evaluation were produced using software provided by SICSTus Prolog.

sociated to the abstract and a field to be filled with comments about the abstract. One of the lists of keywords was the one published with the source document, the other four were randomly selected from the set of 11 remaining keyword lists, they were printed in the form in random order. One page was also available to be completed with comments about the task, in particular with the time it took to the judges to complete the evaluation. We produced three copies of each form for a total of 18 forms.

7.1.2 Subjects
We had a total of 18 human judges or evaluators. Our evaluators were 18 students of the M.Sc. program in Information Science at McGill Graduate School of Library & Information Studies. All of the subjects had good reading and comprehension skills in English. This group was chosen because they have knowledge about what constitutes a good abstract and they are educated to become professionals in Information Science.

7.1.3 Evaluation Procedure
The evaluation was performed in one hour session at McGill University. Each human judge received a form (so he/she evaluated six different abstracts) and an instruction booklet. No other material was required for the evaluation (i.e. dictionary). We asked the judges to read carefully the abstract. They had to decide which was the list of keywords that matched the abstract (they could chose more than one or none at all) and then, they had to associate a numeric score to the abstract representing its quality based on the given criteria. This procedure produced three different evaluations of content and text quality for each of the 36 abstracts. The overall evaluation was completed in a maximum of 40 minutes.

7.2 Results
For each abstract, we computed the average quality using the scores given by the judges. We considered that the abstract indicated the essential content of the source document if two or more judges were able to chose the correct list of keywords for the abstract. The results for individual articles and the average information are shown in Table 4. For a given source document and type of abstract, the value in column

# Art.	Microsoft Abstract		Our Method		S.D. Abstract	
	Indic?	Quality	Indic?	Quality	Indic?	Quality
1	yes	2.66	yes	2.93	yes	4.16
2	no	1.36	yes	3.66	yes	4.00
3	no	1.16	no	3.00	no	4.06
4	yes	3.00	yes	4.00	yes	4.33
5	no	2.16	no	1.76	yes	4.00
6	yes	2.16	yes	4.00	no	4.53
7	no	0.83	yes	2.50	yes	4.40
8	yes	2.33	yes	3.00	yes	4.00
9	yes	2.16	no	2.66	yes	3.66
10	yes	2.16	yes	4.00	yes	3.31
11	yes	2.40	no	2.70	no	4.26
12	no	1.16	no	3.33	no	4.00
Average	70%	1.98	70%	3.15	80%	4.04
Results with Different Documents and Subjects						
Average	80%	1.46	80%	3.23	100%	4.25

Table 4: Results of Human Judgment about Indicativeness and Text Quality

'Indic?' contains the value 'yes' if the majority of the evaluator have chosen the source document list of keywords for the abstract and 'no' on the contrary. The value in column 'Quality' is the average acceptability for the abstract.

Content: In 80% of the cases, the abstracts published with the source documents were correctly classified by the evaluators. Instead, the automatic abstracts were correctly classified in 70% of the cases. It is worth noting that the automatic systems did not use the journal abstracts nor the lists of keywords or the information about the journal.

Quality: The figures about text acceptability indicate that the abstracts produced by Microsoft'97 Summarizer are below the acceptability level of 2.5, the abstracts produced by our method are above the acceptability level of 2.5 and that the human abstracts are highly acceptable.

In a run of this experiment using 30 abstracts from a different set of 10 articles and 15 judges from École de Bibliothéconomie et des Sciences de l'Information (EBSI) at Université de Montréal we have obtained similar results (last row in Table 4).

8 Conclusions

In this paper, we have presented a method of text summarization which produces indicative-informative abstracts. We have described the techniques we are using to implement our method and some experiments showing the viability of the approach.

Our method was specified for summarization of one specific type of text: the scientific and technical document. Nevertheless, it is domain independent because the concepts, relations and types of information we use are common across different domains. The question of the coverage of the model will be addressed in our future work. Our method was designed without any particular reader in mind and with the assumption that a text does have a "main" topic. If readers were known, the abstract could be tailored towards their specific profiles. User profiles could be used in order to produce the informative abstracts elaborating those specific aspects the reader is "usually" interested in. This aspect will be elaborated in future work.

The experiments reported here addressed the evaluation of the indicative abstracts using a categorization task. Using the automatic abstracts reader have chosen the correct

category for the articles in 70% of the cases compared with 80% of the cases when using the author abstracts. Readers found the abstracts produced by our method of better quality than a sentence-extraction based system.

Acknowledgments

We would like to thank three anonymous reviewers for their comments which helped us improve the final version of this paper. We are grateful to Professor Michèle Hudon from Université de Montréal for fruitful discussion and to Professor John E. Leide from McGill University and to Mme Gracia Pagola from Université de Montréal for their help in recruiting informants for the experiments.

References

R. Barzilay and M. Elhadad. 1997. Using Lexical Chains for Text Summarization. In *Proceedings of the ACL/EACL'97 Workshop on Intelligent Scalable Text Summarization*, pages 10–17, Madrid, Spain, July.

G. DeJong. 1982. An Overview of the FRUMP System. In W.G. Lehnert and M.H. Ringle, editors, *Strategies for Natural Language Processing*, pages 149–176. Lawrence Erlbaum Associates, Publishers.

T. Firmin and M.J. Chrzanowski. 1999. An Evaluation of Automatic Text Summarization Systems. In I. Mani and M.T. Maybury, editors, *Advances in Automatic Text Summarization*, pages 325–336.

T.R. Gibson. 1993. *Towards a Discourse Theory of Abstracts and Abstracting*. Department of English Studies. University of Nottingham.

P. Grant. 1992. *The Integration of Theory and Practice in the Development of Summary-Writting Strategies*. Ph.D. thesis, Université de Montréal. Faculté des études supérieures.

H. Jing and K.R. McKeown. 1999. The Decomposition of Human-Written Summary Sentences. In M. Hearst, Gey. F., and R. Tong, editors, *Proceedings of SIGIR'99. 22nd International Conference on Research and Development in Information Retrieval*, pages 129–136, University of California, Beekely, August.

H.P. Luhn. 1958. The Automatic Creation of Literature Abstracts. *IBM Journal of Research Development*, 2(2):159–165.

I. Mani, D. House, G. Klein, L. Hirshman, L. Obrst, T. Firmin, M. Chrzanowski, and B. Sundheim. 1998. The TIPSTER SUMMAC Text Summarization Evaluation. Technical report, The Mitre Corporation.

D. Marcu. 1997. From Discourse Structures to Text Summaries. In *The Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization*, pages 82–88, Madrid, Spain, July 11.

J-L. Minel, S. Nugier, and G. Piat. 1997. Comment Apprécier la Qualité des Résumés Automatiques de Textes? Les Exemples des Protocoles FAN et MLUCE et leurs Résultats sur SERAPHIN. In *1éres Journées Scientificques et Techniques du Réseau Francophone de l'Ingénierie de la Langue de l'AUPELF-UREF.*, pages 227–232, 15-16 avril.

J-L. Minel, J-P. Desclés, E. Cartier, G. Crispino, S.B. Hazez, and A. Jackiewicz. 2000. Résumé automatique par filtrage sémantique d'informations dans des textes. *TSI*, X(X/2000):1–23.

C.D. Paice and P.A. Jones. 1993. The Identification of Important Concepts in Highly Structured Technical Papers. In R. Korfhage, E. Rasmussen, and P. Willett, editors, *Proc. of the 16th ACM-SIGIR Conference*, pages 69–78.

D.R. Radev and K.R. McKeown. 1998. Generating Natural Language Summaries from Multiple On-Line Sources. *Computational Linguistics*, 24(3):469–500.

J. Rowley. 1982. *Abstracting and Indexing*. Clive Bingley, London.

H. Saggion and G. Lapalme. 1998. Where does Information come from? Corpus Analysis for Automatic Abstracting. In *Rencontre Internationale sur l'Extraction le Filtrage et le Résumé Automatique. RIFRA'98*, pages 72–83, Sfax, Tunisie, Novembre 11-14.

H. Saggion and G. Lapalme. 2000. Selective Analysis for Automatic Abstracting: Evaluating Indicativeness and Acceptability. In *Proceedings of the Computer-Assisted Information Searching on Internet Conference. RIAO'2000*, Paris, France, 12-14 April, 2000.