Selective Analysis for Automatic Abstracting: Evaluating Indicativeness and Acceptability

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 have developed a new methodology for automatic abstracting of scientific and technical articles called Selective Analysis. This methodology allows the generation of indicative-informative abstracts integrating different types of information extracted from the source text. The indicative part of the abstract identifies the topics of the document while the informative one elaborates some topics according to the reader's interest. The first evaluation of our methodology demonstrates that Selective Analysis performs well in the task of signaling the topic of the document demonstrating the viability of such a technique. The sentences the system produces from instantiated templates are considered to be as acceptable as human produced sentences.

1 Introduction

Automatic abstracting of textual data can be seen as a process composed of four main steps: (i) the source text is interpreted in order to obtain a "meaning" representation of the source; (ii) the representation is then used to extract the most relevant information which ideally includes (or allows to identify) the topics of the text (main entities, main themes, etc.); (iii) that information is condensed by generalizations and elision of repeated material; and finally (iv) it is presented to the reader in the form of a new text. Most approaches to automatic abstracting concentrate on the first and second steps usually ignoring the last two. Notable exceptions being (Radev and McKeown, 1998; Paice and Jones, 1993). The third step, called condensation, can in some situations be addressed without world knowledge as recent work demonstrates (Barzilay et al., 1999).

In our work, we are focusing on the overall process of automatic abstracting. Although we do not address the issue of generalizations, we are addressing some aspects of elision of repeated information (this is particularly necessary in the type of text we are dealing with). Our method of automatic abstracting called Selective Analysis produces a well known type of abstract: the indicative-informative abstract. This is done in two steps: first, the reader is presented with an indicative text signaling the topics of the document; then, if the reader is interested in additional information about some of the identified topics, the system will present pieces of information from the source document.

[a]Almost all the proposed prototypes rely on a human guided vehicle for solving the first task, while detection and localisation of fruits, which appear to be the most difficult problem, are faced in an automatic mode of operation based on artificial vision. [b]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. [c]The harvester has been tested in laboratory conditions: tests are described and results are given together with some conclusions of the work. [d]Presents the specific mechanical design of the picking arm addressing the reduction of undesirable dynamic effects during high velocity operation. [e]The Agribot 's approach presents a semi-automatic way of operation, with realistic goals, combining harmoniously the human and machine functions. [f]The harvesting strategy that inspires the robotic harvester relies on an operator that will guide the vehicle in the grove and, once stopped, detects the fruits, while the robotic system locates them, plans the picking sequence and makes the approximation and detaching of the fruit. [g]Shows schematic view of the detaching tool and operation and view of the robotic fruit harvester Agribot during laboratory tests.

Topics: Agribot; every target; arm; condition; design; detaching tool; detection; difficult problem; dynamic; fruit; fruit localisation module; ...

Figure 1: Automatic Abstract by Selective Analysis. Source Document: Design and implementation of an aided fruit-harvesting robot (Agribot), R. Ceres, Industrial Robot 25(5), 1998.

In Figure 1, we present an indicative abstract automatically obtained. The source document[1] is a long technical text (36K characters, 5619 words and 220 sentences). The generated abstract which is 4% of the source, is a new text because pieces of information from different parts of the document were extracted and integrated. In order to produce this abstract, all sentences have been automatically interpreted and different types of information have been extracted and merged in order to produce a compact text. Sentences *b, c, d* and *g* of the automatic abstract do not appear in the source document. In particular, sentence *g* was generated by merging information from two different sentences.
The text is presented to the reader along with a list of "topics" (terms) available for expansion, so for example if the reader wants more information about the topic "Agribot" he/she will obtain different types of information such as the text "The Agribot has been developed at Instituto de Automática Industrial to cope with the previously outlined requirements" giving information about the author development.

This kind of abstract could be used in tasks such as accessing the content of the document, deciding if the source is worth reading and obtaining specific types of information about the topics.

If automatic abstracting systems are designed to fulfill those requirements, the generated texts have to be evaluated in function and quality. In this paper, we will address both the evaluation of the function of the indicative abstract and its form. In the following sections we describe our method of automatic abstracting and its implementation, and our evaluation methodology together with the results.

[1]This source document will be used as example all through this paper.

2 Corpus Analysis

We have developed our method of automatic abstracting by studying a corpus of professional abstracts and source documents. Our corpus contains 100 items each composed of a professional abstract and its source (or parent) document. We used as source for the abstracts the journals Library & Information Science Abstracts (LISA), Information Science Abstracts (ISA) and Computer & Control Abstracts. The source documents were found in journals of Computer Science (CS) and Information Science (IS) such as AI Communications; AI Magazine; American Libraries; Annals of Library Science & Documentation; Artificial Intelligence, Computers in Libraries, IEEE Expert, among others (a total of 44 publications where examined).

We manually aligned each sentence of the professional abstract with one or more elements of the source document. This was done by looking for a match between the information in the professional abstract and the information in the source document. The structural parts of the source document we examined are: the title of the source document, the author abstract, the first section, the last section, the subtitles and captions of tables and figures. When the information is not found, we look in other parts of the source document. One alignment is shown in Table 1. The first column contains an identification of the sentence that will be used in the following sections. The second column contains the sentences from the professional abstract. The third column contains the information from the source document. In this particular case, all the information from the 5 sentences of the professional abstract was found in 6 sentences in the introduction of the source document.

We found that in this corpus 72% of the information for the abstracts comes from the following structural parts of the source documents: the title of the document, the first section, the last section and the subtitles and captions. But abstractors not only select the information for the abstract because of its particular position in the source document, they also look for specific types of information which happen to be lexically marked. In Figure 1 the information reported is: **the authors' interests**, **the authors' development**, **the description** of some entities and the explicit mention of **the topics** of the document. This information is lexically marked in the source document by expressions like *we*, *concern*, *here*, *Laboratories*, *develop*, *implement*, *work*, *article*, *discuss*, *overview* among others. Based on this observations we defined a conceptual and linguistic model for the task of text summarization of technical articles.

3 Concepts, Relations and Types of Information

The scientific and technical article is the result of the complex process of scientific inquiry (Bunge, 1967): that starts with the identification of a problem and eventually ends with its solution. It is a complex linguistic record of knowledge referring to a variety of real and hypothetical concepts and relations. Some of them are domain dependent (deceases and treatments in Medical Science, atoms and fusion in Physics; algorithms and proofs in Computer Science) while others are generic of the technical literature (authors, the research article, the problem, the solution, etc.). We have identified 55 concepts and 39 relations, which are typical of a technical article, relevant for the task of identifying types of information for text summarization. This was done by the process of collecting domain independent lexical items and linguistic constructions from the corpus and classifying them using a thesaurus (Vianna, 1980). Afterwards, we expanded the initial set with more valid linguistic constructions not observed in the corpus.

#	**Professional Abstract**	**Source Document**
(1)	The production of understandable and maintainable expert systems using the current generation of multiparadigm development tools *is addressed*.	*We are concerned here* with the production of understandable and maintainable expert systems using the current generation of multiparadigm development tools.
(2)	This issue *is discussed* in the context of COMPASS, a large and complex expert system that helps maintain an electronic telephone exchange.	*GTE Laboratories has developed* COMPASS, a large and complex expert system that helps maintain an electronic telephone exchange.
(3)	As part of the work on COMPASS, several techniques to aid maintainability *were developed* and successfully *implemented*.	As part of our work on COMPASS, *we developed* and successfully *implemented* several techniques to aid maintainability.
(4)	Some of the techniques were new, others were derived from traditional software engineering but modified to fit the rapid prototyping approach of expert system building.	Some of these techniques were new. Others were derived from traditional software engineering, but modified to fit the rapid prototyping approach of expert system building.
(5)	An overview of the COMPASS project *is presented*, software problem areas *are identified*, solutions adopted in the final system *are described* and how these solutions can be generalized *is discussed*.	*This article will overview* the COMPASS project and problem domain, *identify* software problem areas we discovered, *describe* solutions we adopted in the final system, and *discuss* how these solutions can be generalized.

Table 1: Alignment of the Professional Abstract: CCA 58293 (1990 vol.25 no.293) with the Source Document: "Maintainability Techniques in Developing Large Expert Systems." D.S. Prerau *et al.* IEEE Expert, vol.5, no.3, p.71-80, June 1990.

Concepts can be classified in categories such as: the authors (the authors of the article, their affiliation, researchers, etc.), the work of the authors (work, study, etc.), the research activity (actual situation, need for research, problem, solution, method, etc.), the research article (the paper, the paper components, etc.), the objectives (objective, focus, etc.), the cognitive activities (presentation, introduction, argument, etc.). Some of these concepts are presented in Table 2.
Relations refer to general activities of the author during the research and writing of the work: studying (investigate, study, etc.), reporting the work (present, report, etc.), motivating (objective, focus, etc.), thinking (interest, opinion, etc.), identifying (define, describe, etc.). Some of these relations are presented in Table 3. Note that we have identified only a few linguistic expressions used in order to express particular elements of the conceptual model, this is because we were mainly concerned with the development of a general method of text summarization and the task of constructing such linguistic resources is time consuming. Recent works have shown (Minel et al., 2000; Garcia, 1998) that much effort is needed in order to find appropriate linguistic markers and rules for the task of classifying textual segments into semantic categories.

We have identified 52 types of information for the process of automatic text summarization referring to the following aspects of the technical article: background information (situation, need, problem, etc.), reporting of information (presenting entities, topic, subtopics, objectives, etc.); re-

Concept	Explanation & Example	Lexical Items
`author`	The authors of the article. "*I* refer to ..."	*we, I, author, us*
`author related`	Authors' related entity. "The core of *our system* is comprised of..."	*our, my*
`research paper`	The technical article "In *this article*..."	*article, here, paper*
`research`	The research work. "... a broad range of *scientific research*..."	*research, investigation*
`problem`	The problem under consideration "*The lack of* a library severely limits the impact of..."	*difficulty, issue, ...*
`need`	A necessity. "...*the need* for an interface between ..."	*need, necessity, ...*
`acronym`	An acronym "The World Wide Web *(WWW)*..."	Noun Group *(Acronym)*
`paper component`	A component of the research paper. "...some successful applications (*Section 3*)..."	*section, subsection*
`focus`	The general focus. "*A key focus* of the technical specification was ..."	*focus*

Table 2: Some Concepts from the Conceptual Model

ferring to the work of the author (study, investigate, method, hypothesis, etc.); cognitive activities (argue, infer, conclude, etc.); and elaboration of the contents (definitions, advantages, etc.).
Concepts and relations are the basis for the classification of types of information referring to the essential contents of a technical abstract. Nevertheless, the single presence of a concept or relation in a sentence is not enough to understand the type of information it conveys. The co-occurrence of concepts and relations in appropriate linguistic-conceptual patterns is used in our case as the basis for the classification of the sentences. Here we present only a few types of information:

Topic of Document: The author explicitly marks the topic of the document. This is identified in sentences from first or last sections of the document containing verbs of the `make known` relation, and the concepts like `author` and `research paper`.

Ex.: In *this paper we have presented* a more efficient distributed algorithm which construct a breadth-first search tree in an asynchronous communication network.

Relation	Explanation & Example	Lexical Items
`make known`	Introducing the topic of the paper. "In this paper we *present*..."	*describe, present, ...*
`investigate`	Investigating. The phase transition in binary constraint satisfaction problems, i.e...., *is investigated.*	*investigate, ...*
`explain`	Explaining. "The accuracy of a prediction based on the expected number of solutions *is discussed* ..."	*discuss, explain, ...*
`describe`	Describing. "The classical generative planning process *consists of* a search..."	*compose, form, ...*
`advantage`	Identifying advantage. "... simulated annealing and evolutionary programming *outperform* back propagation."	*to have advantage*
`identify`	Characterizing entity. "...a new algorithm *called* OPT-2 for optimal pruning..."	*contain, classify, ...*
`effective`	Identifying effectiveness. "Our algorithm *is effective* for..."	*to be effective, ...*

Table 3: Some Relations from the Conceptual Model

Author Development: The explicit mention of a development of the author. We identify this information by the co-occurrence of the `author` concept and `create` relation.

Ex.: As part of the UK Electronic Libraries programme, *the authors have developed* a simple decision support tool which allows a library manager to compare the total cost of acquiring a given item of information from each of a number of different sources.

Goal of Entity: The explicit mention of the objective of a non conceptual entity. This is marked by the `objective` concept or relation.

Ex.: *The goal* of CCAD is to support exploratory design, while keeping the user central to the design activity.

Description of Entity: An entity is being described. This is identified by the `describe` relation.

Ex.: The algorithm *is based on* dynamic programming.

The types of information are classified in **Indicative** or **Informative** depending on the type of abstract they will contribute to. For example, **Topic of Document** and **Author Development** are indicative while **Goal of Entity** and **Description of Entity** are informative.

3.1 From Source to Abstract

According to Cremmins (Cremmins, 1982), the last step in the production of the summary is the "extracting" into "abstracting" step in which the extracted information will be mentally sorted into a pre-established format and will be "edited" using cognitive techniques, however, he gives little indication about the process of edition. The issue of edition in text summarization (either manual or automatic) have been systematically neglected. In our work, we have partially addressed this by identifying 15 transformations frequently found in our corpus, some of which are computationally implementable. The transformations include among others *syntactic verb transformation* in the domain verbs (like the ones observed in sentence alignments (3) and (5) in Table 1), *conceptual deletion* (like in sentence alignments (1) and (5)), *verb selection* for topic introduction (like in sentence alignment (2) and (5)) and *merge of information* (like in sentence alignment (4)). In our corpus, only 11% of the sentences of the professional abstract were reported exactly as in the original source document. Results of the analysis of this corpus were reported in (Saggion and Lapalme, 1998).

4 Selective Analysis

The implementation of our method relies on the following: the selection of particular types of information from the source text; the instantiation of different types of templates; the selection of some of the templates in order to produce an indicative abstract; the (re)generation of a short but novel text which indicates the topics of the document and; the expansion of the indicative text with topic elaborations. We work with two kinds of templates: **indicative templates** used to organize the information for the indicative abstracts; and **informative templates** that organize the information for the informative abstract. In Table 4, we present the specification of the **Topic of the Document**, **Problem Identification**, **Goal of the Author** and **Definition** templates. 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 re-expression of domain verbs, merging of topical information, re-expression of concepts and acronym expansion. The overall process of automatic abstracting shown in Figure 2 is composed of the following steps (for a complete description see (Saggion, 1999)).

4.1 Pre-Processing and Interpretation

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 their 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

Topic of the Document	
Type:	topic
Id:	integer
Predicate:	instance of make known
Where:	instance of {paper, study, work, research}
Who:	instance of {paper, author, study, work, research}
What:	interpreted string
Position:	section and sentence id
Topic candidates:	list of terms from the What filler
Weight:	integer
Problem Identification	
Type:	problem_identification
Id:	integer identifier
Problem Marker:	instance of problem
Content:	interpreted string
Position:	section and sentence id
Topic candidates:	list of terms from the Content filler
Weight:	integer
Goal of Author	
Type:	goal_of_author
Id:	integer identifier
Goal Marker:	instance of authors' goal
Goal:	interpreted string
Number:	sing or plur
Position:	section and sentence id
Topic candidates:	list of terms from the Goal filler
Weight:	integer
Definition	
Type:	definition
Id:	integer identifier
Topic:	noun group
Predicate:	instance of define
Content:	interpreted string
Position:	section and sentence id

Table 4: Template Specification

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.

4.1.1 Indicative Selection

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. Some indicative and informative patterns are presented in Table 5. 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 zero or more elements of the sentence (conceptual, syntactic and lexical elements match one element while variables match zero or more). The following sentence from the source document "Our goal is to reduce the existing gap

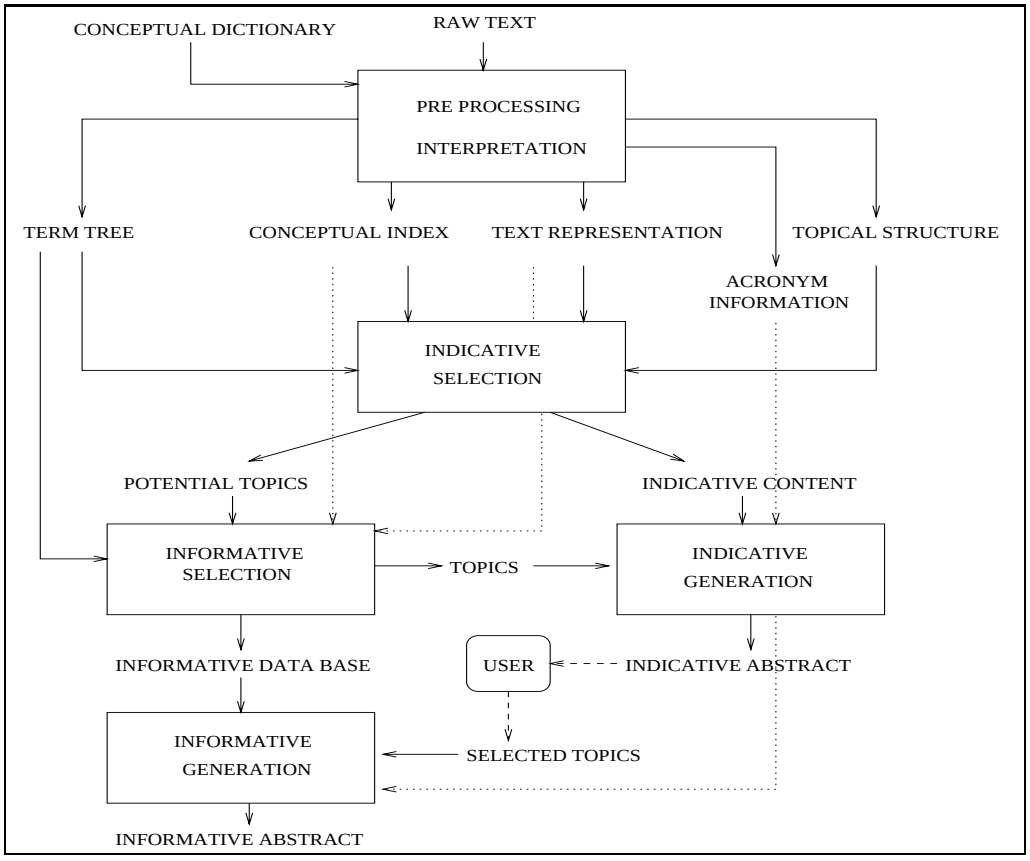

Figure 2: System Architecture

between current manual fruit picking and fully automated harvesting hopefully, reaching this aim in the coming years." matches the pattern **Author s's Goal** and will be used to instantiate a template of type **goal of author**.

For each matched pattern, the system extracts information from the sentence and instantiates a template of type T. For example, the `Goal` slot of the **goal of author** template is instantiated with the string to the right of the `define` relation, the `Content` slot of the **problem identification** template is instantiated with all the sentence (avoiding references, structural elements and parenthesized expressions) while the `What` slot of the **topic of the document** template is instantiated with a string to the left or to the right of the `make known` relation depending on the attribute voice of the verb (active vs. pasive). All the instantiated templates constitute the `Indicative Data Base` (IDB).

The system matches the `topical structure` with the `Topic candidates` slots from the templates in the `IDB`. Two terms $Term_1$ and $Term_2$ match if $Term_1$ is substring of $Term_2$ or if $Term_2$ is substring of $Term_1$ (i.e. *robotic fruit harvester* matches *harvester*). The system selects one template for each term in the `topical structure`: the one with the greatest `Weight` (if more than one, heuristics that consider the type of the template and the position of the sentence used to instantiate the template are applied). The selected templates constitute the `indicative content` and the terms appearing in the `Topic candidates` slots and their expansions consti-

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 5: Pattern Specification

tute 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`.

4.1.2 Informative Selection

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, an informative template will be instantaited, and a link will be created between the topic and the template which will be part of the informative abstract (`informative data base`). For example, the sentence "Since harvesting robots are autonomous systems with self-carried energy sources, it is of paramount importance to reduce as much as possible gravity-related torque to increase the efficiency of the system." matches an informative pattern of definition for the `potential topic` *robot*. The sentence will instantiate a **definition** template which will be included in the `informative data base`.

4.2 Generation

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** of the document, **topic** of sections and **signaling information**. The sorted templates constitute the `text plan`.

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`, `3rd person of singular` in `active voice` at the `beginning of the sentence`; (ii) the interpreted string 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. Some elements in the interpreted string 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 interpreted strings, 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 6. The elaboration of the topics is presented upon reader's demand (informative generation). The information associated with the selected topics is presented in the order of the original text.

Re-Generated Sentences	Sentences from Source Documents
Outlines the developments carried out in robotic systems for hazardous environments in the department.	This paper briefly outlines the developments carried out in robotic systems for hazardous environments in our department.
Describes two non-contact scanning systems, REVERSA and ModelMaker.	Two non-contact scanning systems, REVERSA and ModelMaker, have been described and their application in industry demonstrated.
Shows the RIMHO walking robot and ROBUR arm exchanging gas filter in IUI (Industrializable urban infrastructures) demonstration.	With such geometry the machine can walk with a clearance of 350mm (see Figure 4 The RIMHO walking robot).
	An excellent demonstration (see Figure 6 ROBUR arm exchanging gas filter in IUI demonstration) was held and the IUI system was introduced to the EUREKA officers and to other authorities who declared the project as very successful and promising.
The now-patented solutions were developed at MIT in support of the WAM (the whole-arm manipulator) project.	The now-patented solutions were developed at MIT in support of the WAM project.
Explores the issues and trade-offs that must be considered when designing a system to perform grasping of objects.	In this section we explore the issues and trade-offs that must be considered when designing a system to perform vision-based grasping of objects.
Presents the theoretical model for fettling; and also describes the wrist and experimental setup.	The theoretical model for fettling is presented.
	The wrist and experimental setup is described briefly.

Table 6: Re-Generated Sentences

5 Limitations of the Approach

We implemented our method using 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, on a statistical POS-tagger, on the patterns observed during the analysis of the corpus, and on a process of information extraction. Our prototype only analyses sentences and does not consider relations that cross sentence boundaries. Topic elaboration is only based on term repetition, a common phenomenon in long technical documents. Other cohesive phenomena like anaphora and synonymy were not addressed in the present work and will be subject of

future improvements. Our approach to text generation is based on the regularities observed in the corpus of professional abstracts and so, we are not implementing a general theory of text generation by computers. Our initial corpus of 100 professional abstracts was increased with 100 items mainly in the CS domain in order to validate the model and collect more linguistic patterns, however, the question of completeness remains.

6 Evaluation

We now describe the evaluation of the following aspects of our methodology: indicativeness, how Selective Analysis performs in indicating the essential content of the source document, and acceptability, how adequate the sentences automatically generated are when compared with human sentences.

6.1 Indicativeness

In order to evaluate indicativeness, it is necessary to measure if the automatic system was able to identify the intended "topics" of the source document. This calls for *a priori* knowledge about the topics of the source text which can usually be obtained from a list of keywords or from an "ideal abstract" produced by a human. But an ideal abstract is difficult and costly to produce as some experiences have shown (Kupiec et al., 1995) and this is specially the case when dealing with long technical articles.

Even though work is being done in the evaluation of summarization systems (Mani et al., 1998), there is a clear lack of evaluation resources for the scientific and technical domain. We addressed this by constructing our own evaluation resources with technical articles published on electronic journals on the Web. We use as Gold Standard for evaluation the abstracts published with the source documents (as was the case in (Lin and Hovy, 1997) but for different purposes) and we compared the terms appearing in the automatic abstracts with the terms appearing in the journal provided abstracts. We do not compare sentences with sentences because the abstracts published together with source documents usually contain sentences difficult to match with those of the source document (Teufel and Moens, 1998). The performance of Selective Analysis was measured relative to two other summarization methodologies: abstracts produced using word distribution, and abstracts produced using the commercially available Microsoft Office '97 Summarizer. We implemented the word distribution method by computing the distribution of nouns (common nouns and proper nouns) in all the text (using the result of the POS-tagging process and the canonic form of the words) and then by associating a score to each sentence (the sum of the distribution of its nouns). To produce the abstract, the method chooses top ranked sentences until a compression rate is achieved. This technique, though simple, has been used alone or in combination with other methods in order to produce summaries (Luhn, 1958; Brandow et al., 1995; Kupiec et al., 1995). The two methods used here for comparison purposes were already used in evaluation of text summarization systems and are easily available.

6.2 Experiment 1

For this experiment, we used the text of 25 technical articles found on the Emerald electronic library (Industrial Robot Journal, Internet Research Journal, Assembly Automation Journal, Information Management & Computer Security Journal, and COMPEL Journal) and on the electronic version of the Computer Journal. The articles contain titles, author identification, a short list of keywords (2 to

5), an indication of the type of article (technical, case study, etc.), an abstract, the text of the article, and references. The articles are quite long (from 13K characters to 36K characters with an average of 23K characters). The given abstracts and lists of keywords were not considered in order to produce the automatic abstracts.

We automatically extracted a list of terms from the given abstracts (TAA) using our own resources considering only those terms appearing in the abstract and in the source document. We produced abstracts using Selective Analysis and extracted the list of "topics" accordingly (TSA). We computed the compression ration in number of words[2] for the automatic abstract (CSA) and the abstract provided by the journal (CAA). For our source document example (Figure 1) which contains 5619 words, we had 97% compression for the author abstract (155 words or 3% of the text) and 96% compression for the automatic abstract by Selective Analysis (215 words or 4% of the text). Except for one document, Selective Analysis always produced more verbose abstracts than the provided abstract. The compression ration (between 91% and 96% with and average of 94.4%) was always greater than the compression ratio of 90% used in other summarization evaluations (Mani et al., 1998).

Next, we produced two additional abstracts for each document: one by word distribution and other using the Microsoft Office '97 Summarizer, the compression ratio being the smaller of CSA and CAA (i.e. allowing the other abstracts to be at least as verbose as Selective Analysis). In order to produce the abstract by word distribution, we used the results from the pre-interpretation step in Selective Analysis. In order to produce the abstract with Microsoft Office '97 Summarizer, 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).

Following this, we extracted terms from the abstract obtained by word distribution (TWD) and from the abstract obtained using Microsoft Office '97 Summarizer (TM). We used the very same techniques than in selective analysis (i.e. we interpreted the sentences in both abstracts identifying noun groups and extracting terms). The terms in TSA, TWD and TM were compared with the terms in TAA and recall, precision and F-score measures were calculated for the three methodologies and each individual source document. The measures were computed using the following formulas:

$$Recall(Method) = \frac{\|TAA \bigcap TMethod\|}{\|TAA\|}$$

$$Precision(Method) = \frac{\|TAA \bigcap TMethod\|}{\|TMethod\|}$$

$$Fscore(Method) = \frac{2*Recall(Method)*Precision(Method)}{Recall(Method)+Precision(Method)}$$

where $Method$ is SA, WD or M.

In Table 7, we show the terms extracted from the four abstracts of the source document presented in Figure 1, the topics correctly identified by each method in bold and the three measures.

[2]For this experiment we do not use character compression because Microsoft Summarizer works based on word compression.

6.3 Results of Indicativeness

In Table 8, we present the figures obtained for the 25 articles and the three methodologies and the average information. These numbers indicate that Selective Analysis performs better than word distribution and Microsoft Office Summarizer when the source document is a technical article and the compression ration is high (more than 91%). There is indication that Selective Analysis performs better in precision with a gain of 125% over the two other methods, performing better than the other two methods in 20 cases. This is due to the fact that the terms produced by our method are those additionally elaborated in the source document and not only "mentioned" in the indicative abstract. Although the average recall for the 25 articles indicate a gain of 25% over word distribution and Microsoft Office '97 Summarizer, there is no clear indication of better performance in general (Selective Analysis performed better than the other methods in 10 cases, word distribution in 5 cases and Microsoft Summarizer in 7 cases). Regarding F-score, we have obtained a gain of 85% over word distribution and a gain of 84% over Microsoft Summarizer.

Source	Terms
TAA	accuracy; computer; dependence; detection; fruit; harvester; laboratory test; motion; operator; picking arm; repeatability; result; robotic harvester; sequence; specific design; unstructured environment; and work.
TSA R .55 P .22 F .31	Agribot; every target; arm; condition; design; detaching tool; **detection**; difficult problem; dynamic; **fruit**; fruit localisation module; function; grove; **harvester**; human guided vehicle; integration; laboratory; laboratory condition; **laboratory test**; laser telemetry; localisation; operation; **operator**; **picking arm**; picking sequence; problem; **result**; robot; robotic fruit harvester; **robotic harvester**; robotic system; schematic view; system; task; test; tool; vehicle; velocity; way; and **work**.
TWD R .35 P .09 F .14	Agribot; IR; angle; approximate idea; approximation movement; associate strategy; board display; cabin; color; **computer**; condition; configuration; control; control board; data acquisition; data processing; **detection**; device; different module; distance; electrical brake; encoders; evaluation; external device; freedom pan; **fruit**; fruit localisation system; fruit location; function; general function; image; incidence; inductive limit switch; joystick; laser beam; laser beam direction; laser telemeter; localisation module; location calculation; measure; motor reference; **operator**; orientation; **picking arm**; pneumatic valve; pointer coaxial; pressure sensors; quality; range measurement; robotic fruit harvester; robotic system; see; **sequence**; standard deviation; stereoscopic vision; surface; system; target; testing; tilt mechanism; tool; trajectory determination; triangulation technique; two camera; and two-degree.
TM R .47 P .16 F .16	Agribot; Agribot picking arm; Localisation result harvester arm performance; acceleration figure; aided fruit-harvesting robot; arm; **computer**; configuration; design; detaching tool; **detection**; dynamic data structure; environment; **fruit**; fruit depth distribution; fruit height distribution; fruit localisation; fruit-picking zone; harvester mechanical structure; implementation; joystick; **laboratory test**; laser range-finder testing setup; laser spot; localisation; localisation module; location; manipulator; mechanical design; new approach; **operator**; parallelogram structure; **picking arm**; pointing and picking processes; presented work; remarkable advantage; **result**; robot; robotic fruit harvester; robotic system fruit localisation module; schematic representation; **sequence**; specification; spherical coordinate; statistical model; target fruit; testing; tool motor; trajectory; velocity; and view.

Table 7: Terms Extracted from the four Abstracts and Recall (R), Precision (P) and F-score (F)

Article Number	Selective Analysis			Word Distribution			Microsoft Summarizer		
	Rec.	Prec.	F-score	Rec.	Prec.	F-score	Rec.	Prec.	F-score
1	**.29**	**.25**	**.27**	.14	.06	.08	.14	.05	.05
2	**.27**	**.36**	**.31**	.13	.07	.09	**.27**	.11	.16
3	0	0	-	**.12**	**.02**	**.03**	0	0	-
4	**.50**	**.25**	**.33**	.17	.04	.06	0	0	-
5	.30	**.23**	.26	.17	.09	.12	**.43**	**.23**	**.30**
6	**.33**	**.18**	**.23**	.17	.06	.09	.28	.07	.11
7	.40	**.36**	**.38**	**.50**	.19	.28	.10	.07	.08
8	**.14**	**.08**	**.10**	0	0	-	**.14**	.02	.03
9	**.53**	**.22**	**.31**	.35	.09	.14	.47	.16	.16
10	.40	.09	.15	**.60**	**.11**	**.19**	.20	.04	.04
11	**.25**	**.06**	**.10**	**.25**	.05	.08	**.25**	.04	.04
12	.27	**.09**	.13	.18	.05	.08	**.45**	**.09**	**.15**
13	.19	**.20**	**.19**	.44	.16	.23	**.25**	.09	.13
14	**.37**	**.35**	**.36**	.33	.19	.24	.20	.18	.18
15	**.50**	**.15**	**.23**	0	0	-	.25	.04	.04
16	.11	.07	.09	**.44**	**.13**	**.20**	.33	.07	.07
17	**.25**	**.09**	**.13**	.12	.02	.03	.12	.02	.03
18	.29	**.21**	**.24**	0	0	-	**.43**	.09	.15
19	.09	.08	.08	.09	.04	.06	**.36**	**.17**	**.17**
20	.27	.40	.32	**.68**	**.43**	**.53**	.14	.12	.13
21	.29	**.18**	.22	.36	.12	.18	**.43**	.16	**.23**
22	.13	**.13**	**.13**	.13	.05	.07	**.20**	.09	.12
23	**.29**	**.24**	**.26**	.14	.04	.06	.21	.07	.10
24	**.33**	**.25**	**.25**	.17	.03	.03	.17	.04	.04
25	**.57**	**.20**	**.20**	.29	.06	.06	.29	.08	.08
Average	**.29**	**.19**	**.22**	.24	.08	.12	.24	.08	.12

Table 8: Detailed Recall , Precision and F-score for the 25 Technical Articles and Average Information across Documents

6.4 Acceptability

In this work, we only address the issue of sentence acceptability. In order to evaluate the acceptability of the sentences produced using our method, we used human judges and we asked them to decide if the sentences produced by our system are acceptable to be included in indicative abstracts when compared with human produced sentences.

6.5 Experiment 2

In this experiment, we used 3 human judges with experience in reading technical articles. We presented the judges with a list of 150 randomly selected sentences from three different sources: (1) 50 sentences written by professional abstractors, (2) 50 sentences written by the authors of source documents which contain the information reported in the professional abstracts of our corpus or in the abstracts we generate, and (3) 50 sentences produced by our system. In Table 9, we show one

sentence of each type. The sentences were presented in random order and without source indication. We asked the judges to decide for each sentence if it was acceptable or not to be included in indicative abstracts. The sentences had to be judged independent one another. As in (Coch, 1996), we give the judges some criteria for sentence acceptability such as "good grammar" and "correct spelling" and also a short statement "the sentences are generally brief, and usually, don't contain references to the source document." We gave the judges two examples of good indicative abstracts written by professional abstractors. The judges were informed that they could consider a sentence acceptable even if it contained minor errors. They were also told that some acronyms could appear without expansion and that this situation was also acceptable (this will be an issue once we evaluate text acceptability). We used the vote of the majority in order to consider a sentence as acceptable.

SA	Presents the architecture of the agent; describes its design and implementation; and gives some examples showing the cluster labels generated by the clustering algorithm.
PA	Presents a more efficient Distributed Breadth-First Search algorithm for an asynchronous communication network.
SD	The software presented in this article adds new motion features to the Aria-Delta parallel robot.

Table 9: Sentences from the 3 Sources: Selective Analysis (SA), Professional Abstractor (PA), and Source Document (SD).

6.6 Result of Acceptability

The results of the experiment are presented in Table 10. These indicate a good acceptability rate for Selective Analysis when compared with human generated sentences. Most of the sentences automatically generated were unacceptable for the very same reasons that human produced sentences were unacceptable (too brief, too long, use of passive voice, impersonal, etc.). The sentences produced by professional abstractors were always more acceptable than the other two types of sentences. Note that the information from the source documents comes from different structural elements including titles and captions, and that explains in part the results.

Source	Judge 1	Judge 2	Judge 3	Accepted
Selective Analysis	42 (84%)	48 (96%)	22 (44%)	42 (84%)
Professional Abstractor	46 (92%)	50 (100%)	29 (58%)	47 (94%)
Source Document	37 (74%)	48 (96%)	25 (50%)	38 (76%)

Table 10: Number of Acceptable Sentences and Average Acceptability

7 Conclusions

In this paper, we have presented a method of technical text summarization called Selective Analysis. The method is based on the superficial analysis of the text, the instantiation of templates with specific types of information, the presentation of the information in an indicative abstract that introduces the

topics of the document, and the expansion of the topics according to the readers' interests.

We have evaluated two aspects of the automatic abstracts: indicativeness and sentence acceptability. The evaluation of indicativeness consisted in comparing the "topics" computed by three different methods with the "topics" from author abstracts using recall and precision measures. On the overall, Selective Analysis performed better that the other methods but there is no indication of better performance in a majority of the cases. Regarding sentence acceptability, we found that sentences produced automatically from some types of instantiated templates have quality comparable to human produced sentences.

Other works have addressed the problem of automatic abstracting of scientific and technical papers including (Paice and Jones, 1993) for one specific technical domain and (Lehmam, 1997) for French articles. While it would be impossible to compare our approach with theirs because our approach is domain independent and for English texts, it would be interesting to compare our approach with other theoretical or practical methods recently developed.

Even if we managed to evaluate our method through resources which can be found on the Web, important questions remain. We have chosen to use the abstract provided with the source document as an ideal abstract. But sometimes those abstracts fail to indicate the essential content of the document. In fact, we had to exclude some articles from our test set because the terms appearing in the provided abstracts were not found in the technical document and as a result all the three methods failed to indicate the "topics" of the source text. But, the fact that a term appearing in the provided abstract didn't appear in the source document doesn't mean that it is not a topic. That term could be obtained using a deductive process (for example the term "pet" can be obtained from generalization of "dog" and "cat"). Unfortunately, the methodologies presented here cannot produce new terms from those explicitly found in the source text. Based on those observations and on the fact that we have only addressed the acceptability of sentences out of context, we have carried out other evaluations with human judges (evaluators) in order to assess indicativeness (using human produced keywords as content indicators) and text quality. In those experiments, the abstracts produced by Selective Analysis were compared with human produced abstracts and with other summarization methodology in the task of text classification. We found that the abstracts by Selective Analysis indicated the content of the source document and the evaluators considered the abstract produced by our method to be of acceptable quality (Saggion and Lapalme, 2000).

Selective analysis was designed to produce a very short abstract (less than 10% of the source) using natural language re-generation techniques and allowing the reader access the content of the document, this is an unusual case in automatic abstracting. Regarding content evaluation, we have only addressed the issue of indicativeness. Informativeness will be the subject of our next work: we will evaluate how Selective Analysis performs in the task of informing a reader interested in knowing more about the topics of the document.

Acknowledgments

We would like to thank two anonymous reviewers for their comments which helped us improve the final version of this paper, and the evaluators for their participation in the second experiment. The first author is supported by Agence Canadienne de Développement International (ACDI) and Fundación Antorchas (A-13671/1-47), Argentina. He was previously supported by Ministerio de Educación

de la Nación de la República Argentina (Resolución 1041/96) and Departamento de Computación, Facultad de Ciencias Exactas y Naturales, UBA, Argentina.

References

Barzilay, R., McKeown, K., and Elhadad, M. (1999). Information Fusion in the Context of Multi-Document Summarization. In *Proceedings of the ACL'99*, pages 550–557, Maryland, USA.

Brandow, R., Mitze, K., and Rau, L. (1995). Automatic Condensation of Electronic Publications by Sentence Selection. *Information Processing & Management*, 31(5):675–685.

Bunge, M. (1967). *Scientifc Research I. The Search for System.* Springer-Verlag New York Inc.

Coch, J. (1996). Evaluating and Comparing three Text-production Techniques. In *COLING-96, The 16th International Conference on Computational Linguistics*, volume 1, pages 249–254, Copenhagen, Denmark.

Cremmins, E. (1982). *The Art of Abstracting.* ISI PRESS.

Garcia, D. (1998). *Analyse automatique des textes pour l'organisation causale des actions. Realisation du système informatique COATIS.* PhD thesis, UFR: Institut des Sciences Humaines Appliquées (ISHA). Université de Paris-Sorbonne (Paris IV).

Kupiec, J., Pedersen, J., and Chen, F. (1995). A Trainable Document Summarizer. In *Proc. of the 18th ACM-SIGIR Conference*, pages 68–73.

Lehmam, A. (1997). Une structuration de texte conduisant à la construction d'un système de résumé automatique. In *Actes des Journées Scientifiques et Techniques du Réseau Francophone de l'Ingénierie de la Langue de l'AUPELF-UREF*, pages 175–182.

Lin, C. and Hovy, E. (1997). Identifying Topics by Position. In *Fifth Conference on Applied Natural Language Processing*. Association for Computational Linguistics.

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

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

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

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

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

Saggion, H. (1999). Using Linguistic Knowledge in Automatic Abstracting. In *Proceedings of the ACL'99*, pages 596–601, Maryland, USA.

Saggion, H. and Lapalme, G. (1998). Where does Information come from? Corpus Analysis for Automatic Abstracting. In *RIFRA'98. Rencontre Internationale sur l'extraction le Filtrage et le Résumé Automatique*, pages 72–83.

Saggion, H. and Lapalme, G. (2000). Evaluation of Content and Text Quality in the Context of Technical Text Summarization. Submited.

Teufel, S. and Moens, M. (1998). Sentence Extraction and Rhetorical Classification for Flexible Abstracts. In *Intelligent Text Summarization*, pages 16–25.

Vianna, F. d. M., editor (1980). *Roget's II. The New Thesaurus.* Houghton Mifflin Company, Boston.