Automatic Summarization: An Overview

Horacio Saggion

In Revue française de linguistique appliquée 2008/1 Vol. XIII , pages 63 to 81
Publisher Publications linguistiques

ISSN 1386-1204
DOI 10.3917/rfla.131.0063
Uploaded: 06/20/2008

Rev. franç. de linguistique appliquée, 2008, XIII-1 (63-81)

Automatic Summarization: An Overview

Horacio Saggion
University of Sheffield

Abstract: *A summary is a condensed version of a document. It contains the most relevant information in context found in the source document. Automatic summarization is the process of producing text summaries by computer. Although research on automatic summarization started in the late 50s, the increasing volume of electronic text and recent international evaluation efforts have fuelled research in this field. The paper gives an overview of basic concepts in automatic text summarization together with examples of available tools, systems, evaluation, summarization experiments, summarization in practical settings, and discusses the role of linguistic information in the summarization task.*

Résumé : *Un résumé est un texte concis qui rend compte du contenu essentiel d'un document par rapport à une tache déterminée. Cet article donne un aperçu de la recherche en résumé automatique – la tâche qui vise à produire des résumés par ordinateur. Bien que les recherches en résumé automatique aient débuté dans les années cinquante, il y a eu récemment un très fort renouveau de l'intérêt dans ce domaine en raison de la quantité des textes disponibles en format numérique et en réaction aux efforts d'évaluation internationale. On introduit ici la notion de résumé automatique et les techniques utilisées pour les produire. On présente des systèmes de résumé automatique disponibles aussi bien que des applications réels du résumé. On discute le rôle des connaissances linguistiques dans le processus automatique et le problème de l'évaluation.*

1. Introduction

Text summarization or abstracting has always been a key activity in the information access context. Document summaries provide readers with condensed versions of the most relevant information found in documents, they can therefore help readers assess the value of the document without having to read it, or can be used as content repositories for extracting valuable facts or information. Still today and in spite of the Internet revolution and the easy access to full documents, summaries are used as valuable document surrogates; they are notably used by abstracting services or abstracting databases to inform the research community about advances in science; in occasions and because of copyright issues, an abstract of a scientific publication is the only freely available piece of information we have access to about a piece of research. A special type of document summary, the "headline" is one of the main means to decide whether or not to read a full article in a newspaper or news web site.

It is almost impossible to talk about summarization without mentioning first a number of types of summaries and summarization tasks recognised by the information science

(Cremmins 1982 ; Lancaster 2003) and computational linguistic communities (Spärck Jones 1999). Where transformation of the source document is concerned two types of summaries can be considered: the *extract* is a summary containing passages selected from the source document (usually sentences); the *abstract* is a summary where the information from the source document has been transformed in some way. Where function that the summary serves is concerned two main summary types are usually recognised (Rowley 1982): an *indicative* summary alerts the reader about the contents of the document while an *informative* summary provides quantitative and qualitative information from the source document. Summaries may be produced for one document (single-document summarization) or for a set of *related* documents (multi-document summarization). Summaries can be produced in the language of the source documents (mono-lingual summarization) or in another language (cross-lingual summarization), a useful situation when a user wants a brief account of what is published in a language (s)he does not understand.

The selection of material for a summary can be influenced by a particular information access interest such as the need for summaries where the information would match a particular topic (topic-based summarization) (e.g., the summary should describe advantages or disadvantages of method X when applied to problem Z) or where the summary would contain information answering specific questions (e.g. what is the problem studied? what are the main results?). Various forms of linguistic knowledge have informed text summarization algorithms. Linguistic knowledge in text summarization has taken the form of specialized lexicons of cue words or expressions, semantic rules designed to identify key domain information, coreference and cohesion knowledge, and rhetorical information. It is worth noting that linguistic knowledge is also present in summarization algorithms when the system uses morphological information or parts-of-speech tagging or terminology detection, etc., thus no summarization process is free from linguistic knowledge.

Automatic text summarization is a technology to produce machine-generated summaries. Although work in this area started in the late 50s (Luhn 1958), the development of the Internet, the availability of massive textual databases together with international evaluation efforts such as the Document Understanding Conferences - DUC (Over & *al*. 2007) or the Translingual Information Detection Extraction and Summarization programme in the US or the Text Summarization Challenge (Okumura & *al*. 2004) in Japan have fuelled research in this field.

Even with the many advances in information technology, summarization is still a field where improvements are much needed, e.g. although automatic systems are outperforming simple baselines their performance is still behind that of humans. Yet producing summaries by hand is a very costly and time consuming process which could be facilitated by the incorporation of natural language processing technologies, at least in some stages of the production process. A considerable number of events have taken place in the last two decades starting with the seminal 1993 Dagstuhl Seminar on Text Summarization for Intelligent Communication (Endres-Niggemeyer & *al.* 1993) through the publication of relevant special issues on the topic (Spärck Jones & Endres-Niggemeyer 1995; Radev & *al*. 2002; Harman 2007), and the organization of workshops (Mani & Maybury 1997; Radev & Hovy 1998; Saggion & Minel, 2005) and book publications (Mani & Maybury 1999; Mani 2001). Nowadays, automatic text summarization is found in many software solutions: *Google* provides summaries containing query keywords; *Microsoft Word* provides the Autosummarize option. InXight (http://www.inxight.com), Pertinence Summarizer (http://www.pertinence.net), and Copernic (http://www.copernic.com) are only a few examples of commercial summarization solutions.

This paper will give an overview of basic techniques used in text summarization; describe summarization evaluation; present experiments on content selection; and summarization in practical settings.

2. What information should be included in a summary

There are two main problems in text summarization: one is the problem of selecting the most relevant information from a source document or documents, the second problem is how to express that key information in the final summary. A compression parameter is usually specified as part of the process, this can be an absolute number of words to be produced or a percent of the original text. One of the main problems with these tasks specifications is that they are too open-ended, and this openness sometimes does not help the development of satisfactory summarization solutions. Usually selection of information depends not only on the document to be summarized but also on contextual factors such as the audience (e.g., does the reader need background information?) or the tasks the user/reader has to carry out with the summary (e.g., indexing, classification, question answering). When minimal requirements are specified in a summarization task (e.g. summarize this document in 200 words), we talk about generic summarization meaning that the perspective of the author of the document would be taken into account when deciding on what content to select from the source.

Most attention has been paid to the problem of what to extract from the source document, however research also has been carried out on how to create well formed summaries.

Approaches to the first problem have tried to come up with a list of relevant features that are believed to indicate the relevance of a sentence in a document or set of documents (See Mani 2001). These features can be used alone or in combination to produce sentence (or paragraph) relevance scores which in turn are used to rank sentences/paragraphs and select the top ranked ones (up to a certain compression) as the resulting summary content, in cases the scoring is some sort of probability which indicates the likelihood that the sentence belongs to a summary. Various interpretations exist of how to compute the features.

The simplest strategy for presenting the summary is a concatenation strategy which outputs the selected sentences in order of occurrence in the document. This has the obvious disadvantages of producing fragmentary texts exhibiting problems of cohesion and coherence (Paice 1990).

Some features investigated in the literature include the use of frequency information (term frequency or tf) combined with inverted term frequency (or idf) from corpus statistics. This feature implements the idea that the more frequent (and at the same time more discriminative) a term is in a document the more relevant the term is and as a consequence it is worth selecting sentences containing the term (Luhn 1958).

The position of a sentence in the document has also informed content selection strategies (Baxendale 1958; Edmundson 1969; Lin & Hovy 1997), in the news and because of its pyramidal discourse structure relevant/new information (the key events) is usually expressed in the leading paragraph, and therefore sentences from the leading paragraph are considered relevant. In scientific papers, sections such as introduction and conclusion report on the objectives and findings of the research, so sentences contained in those sections can be considered as important sections to find statements about research objectives and findings of a piece of research.

Because titles of articles, especially scientific ones, sometimes state the theme or subject dealt within the article, then one may consider that sentences containing terms from the title are relevant for a summary (Edmundson 1969).

The presence of specific formulaic expressions, in particular in the scientific domain but also in the legal domain, has also been considered a useful cue to find information relevant for a summary (Paice 1981). Formulaic expressions such as "This paper presents…", "Our results indicate…", "We conclude that…" are used by paper authors to explicitly introduce key information. In fact, this is also a strategy used by professional abstractors to produce abstracts acknowledged in abstract writing studies (Endres-Niggemeyer & *al*. 1995; Cremmins 1982).

In the context of creating a summary of multiple documents one feature usually exploited is the proximity in content of a sentence to the centroid of the set of documents to be summarized (Radev & *al.* 2000) which is a set of words in a cluster of documents considered statistically important.

Deep linguistic information has been used in many approaches. Saggion and Lapalme (2002) use a conceptual dictionary (where words are associated with specific concepts) and a set of conceptual/linguistic patterns implemented as regular expressions in order to select information for the summary; the approach outperforms superficial methods. It is worth noting that for specific domains dictionaries or lists of cue-words can be semi-automatically constructed as it is a set of extraction rules. Barzilay and Elhadad (1997) use the *WordNet* lexical database for the computation of lexical chains; their approach selects sentences containing members of strong chains. Marcu (1997) applies rhetorical parsing and demonstrates that nuclear information in a rhetorical tree computed automatically correlates with the idea of relevant information in humans.

These are only a few of the indicators or approaches which may be used by a summarization algorithm in the selection of sentences or passages for a summary.

3. Evaluating Content

Evaluation of summaries is a complex issue and in the last decade much research has been dedicated to this problem. While research in text summarization has always presented evaluation results of one type or another, ten years ago the first system independent evaluation the SUMMAC evaluation took place (Mani & *al.* 2002). This event was very significant for the research community because for the first time systems were compared and measured using the same yardstick.

Two types of evaluation are generally considered in summarization (Spärck Jones & Gallier 1995). In an *intrinsic* evaluation the summaries produced are evaluated in terms of whether they contain the main topics of the source and whether they are acceptable texts. Variables measured can be the proportion of key information covered or precision/recall statistics (see later) as well of grammaticality and coherence scores. The content of an automatic summary can be compared to the content of a human produced summary in terms of word or "propositional" overlap. In an *extrinsic* evaluation, the summaries are evaluated in a concrete task seeking to verify if the summaries are instruments which could be used instead of full documents in specific situations. Variables measured can be accuracy in performing a task and time to complete the task. While extrinsic evaluation is very attractive from the point of view of information access, it is also much time consuming and costly, making its implementation limited. Current high scale evaluation of summarization systems is mainly carried out under the auspices of the National Institute for Standards and Technology with the Document Understanding Conferences programme (Over & *al.* 2007) which is mainly intrinsic, although some efforts to simulate specific information access tasks are also implemented.

In the context of single document summarization, metrics for evaluating sentence extraction systems are precision and recall (Firmin & Chrzanowski 1999). Precision is ratio of the number of summary-sentences identified by the system to the number of sentences. Recall is the ratio of summary-sentences identified by the system to the number of true summary sentences. Precision and recall have been used in the past and are nowadays somehow resisted by researchers because they somehow fail to measure content coverage – they only take into account the identity of a sentence and not its content. However and in spite of their limitations they serve as the basis for current "content-based" fine-grained metrics adopted by the research community such as ROUGE (Lin & Hovy 2003), which is in fact a recall metric which considers n-grams as units for comparing automatic and human summaries.

When intrinsic evaluation is applied, an automatic summary is usually compared to a set of summaries produced by humans; depending on the evaluation setting different human informants have been used from information analysts as carried out in DUC (Over & *al.* 2007) to students or experts in abstract writing (Saggion & Lapalme 2002). It is a striking fact that humans do not agree in what information needs to be included in a summary. Some studies show agreement as little as 46% when two humans are asked to select the most important paragraphs of an article (Salton & *al.* 1997). Other studies showed that humans tend to agree more in what the *most* important content is, rather than in *all the important* content. There are however methods to palliate for the low agreement among humans. The pyramid evaluation method (Nenkova & Passonneau 2004) tries to address this issue by considering multiple ideal summaries and by considering the distribution of the information in the set of summaries, an information or fact which is observed in all human summaries is considered more important than a fact observed in a single summary; system summaries are scored accordingly.

Current summarization evaluation rely on so called "content-based" metrics (Saggion & *al.* 2002a), which compare the content of an automatic summary to the content of a human summary producing a similarity score. A high score indicates that the automatic summary is close in content to the human summary (e.g., a sentence such as "Three people were killed in the blast" and "In the blast three were killed" should be considered close in content in spite of their differences). Some metrics used are word or lemma overlap, cosine similarity which treats documents as vectors of terms, or longest common subsequence between sentences which takes into account the number of minimal transformations to transform one sentence into another. The ROUGE package implements a number of metrics based on n-gram comparison. However, and in spite of having achieved correlation with human judgement on content, it is unclear how good they are to capture semantic similarity in order to be used in the evaluation of non-extractive summaries.

4. Summarization Tools

Various summarization packages are available, a well known system is MEAD developed at the University of Michigan (Radev & *al.* 2004). The system is a publicly available toolkit for multi-lingual summarization and evaluation. Various algorithms for feature computation are implemented and include position-based, centroid-based, term frequency, and query-based summarization. The toolkit also includes evaluation methods: co-selection (precision and recall), relative-utility (Radev & *al.* 2003), and content-based metrics. The methods can be combined to produce a sentence score which is used as the basis for ranking and extracting sentences. A redundancy removal program makes sure that sentences similar to previously selected sentences are not included in the summary - this is used in the context of multi-

document summarization where one expects repeated information to appear in different sources.

A second set of algorithms for computing summarization features also freely available is the SUMMA summarization toolkit developed at the University of Sheffield (Saggion 2002) which is implemented as a set of GATE (General Architecture for Text Engineering) processing and language resources (algorithms and data).

GATE is a framework for the development and deployment of language processing technology in large scale (Cunningham & *al.* 2002). It provides three types of resources: Language Resources (LRs) which collectively refer to data; Processing Resources (PRs) which are used to refer to algorithms; and Visualisation Resources (VRs) which represent visualisation and editing components. GATE can be used to process documents in different formats including plain text, HTML, XML, RTF, and SGML. The documents in GATE contain one or more annotation sets. Annotations are generally updated by PRs during text processing. Each annotation belongs to an annotation set and has a type, a pair of offsets (the span of text one wants to annotate), and a set of features and values that are used to encode various types of information. Features (or attribute names) are generally strings. Attributes and values can be specified in an annotation schema which facilitates validation and input during manual annotation. Programmatic access to the annotation sets, annotations, features and values is possible through the GATE Application Program Interface (API). Some typical components of GATE are a tokeniser, a sentence splitter, a parts-of-speech tagging process, and a named entity recognition module.

SUMMA uses some of the default GATE components and also uses the document to store computed values (creation of features and values and special annotations). Summarization components can be combined by the user in a customisable application. A "vanilla" kind of system based on the combination of a subset of features (e.g., position, title, frequency) is also provided which allows the creation of genetic summaries. The objective of the tools is to provide an adaptable tool for the development, testing and deployment of customisable summarization solutions. Processing resources compute numeric features for each sentence in the input document which indicates how relevant the information in the sentence is for the feature. The computed values are combined in a linear formula to obtain a score for each sentence which is used as the basis for sentence selection. Sentences are ranked based on their score and top ranked sentences selected to produce an extract. The tool can be used in the GATE user interface or in a standalone program. The features can also be used in order to train a sentence classification program and the trained system used to produce summaries for unseen document (this is detailed below). An example summary obtained with the tool can be seen in figure 1.

In the middle pane of the figure, we show a document from the Summbank corpus (see below) which has been highlighted by the summarization toolkit with the most important sentences. The sentences have been scored using features to be described below. In the bottom part of the figure the scores associated to each sentence are displayed. The highlighted sentences are annotations stored in an annotation set in the GATE document. These sentences can be exported to a text file using a component provided with the summarization toolkit.

A corpus statistic module computes token statistics including term frequency - the number of times each term occurs in the document (tf). The vector space model has been implemented and it is used to create vector representations of different text fragments - usually sentences but also the full document. Each vector contains for each term occurring in the text fragment, the value tf*idf (term frequency * inverted document frequency). The inverted document frequency of a given term is the number of documents in a collection containing the term.

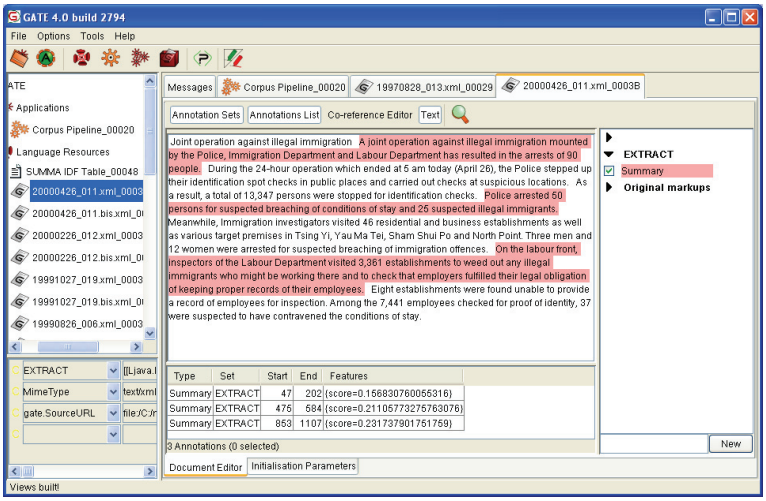

Figure 1. *Summary Computed by the SUMMA toolkit in the GATE GUI.*

These values can be loaded into the system from a table or can be computed on the fly by the summarization tool (local idf values). With the latter option the values can be then be saved for future use.

The term frequency module computes the sum of the tf*idf of all terms in each sentence – note that because frequent terms such as 'the' have close to zero idf value, then their contribution to the term frequency feature is minimal. These values are normalised to yield numbers between 0 and 1. In a similar way, a named entity scorer module computes the frequency of each named entity in the sentence. This process is not based on the frequency of named entities in a corpus but on the frequency of named entities in the input document. A named entity occurring less frequently is more valuable than a named entity observed across different sentences.

A content analysis module is used to compute the similarity between two text fragments in the document represented in the vector space. The measure of similarity is the cosine of the angle between the two vectors. These values can be stored as sentence features and used in the scoring formula. There are various ways in which we use this similarity computation, one is to compute the similarity between the title of the document to each sentence (title method), another one is to compute the similarity of each sentence to a particular user query (query-based method), yet another is to compute the similarity of each sentence to the first sentence of the document, etc.

The sentence position module computes two features for each sentence: the absolute position of the sentence in the document and the relative position of the sentence in the paragraph. The absolute position of sentence i receives value i^{-1} while the paragraph feature receives a value which depends on the sentence being in the beginning, middle or end of paragraph - these values are parameters of the system.

For a cluster of related documents, the system computes the centroid of the set of document vectors in the cluster. The centroid is a vector of terms and values which is in the centre of the

cluster. The value of each term in the centroid is the average of the values of the terms in the vectors created for each document.

The similarity of each sentence in the cluster to the centroid is also computed using the cosine metric. This value is stored as a sentence feature and used during sentence scoring in multi-document summarization tasks (Saggion & Gaizauskas 2004).

The adaptable components described here have been used to create different summarization applications including single multi-lingual summarization (Demetriou & *al.* 2004), cross-lingual summarization (Saggion 2006), topic-based summarization (Saggion 2005), and multi-document summarization (Saggion & Gaizauskas 2004).

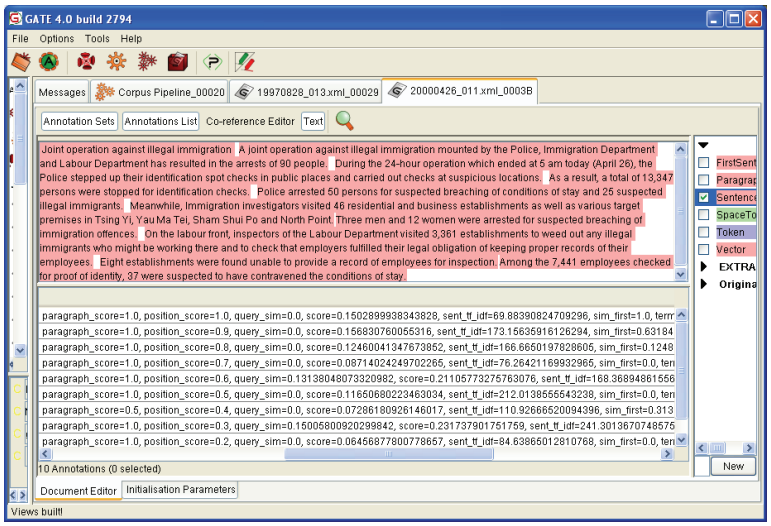

Figure 2. *Features Computed by the SUMMA toolkit displayed in the GATE GUI.*

Figure 2 shows the document annotated with different summarization features and the score produced by the summarization toolkit. Features are stored for each sentence in the document.

5. Learning Summary Informational Content Experiments

In the last decade there has been a great interest in the development of adaptable summarization solutions because of the time and cost usually associated with the manual development of rules for content identification. Summary content can be automatically learned provided that one has access to annotated data and that this is regular enough. Annotated data may consist of documents which have been annotated with the sentences considered relevant by a human or set of humans, e.g. a set of pairs <document, extract>. It may also consist of pairs of <document, abstract> which can be automatically transformed into <document, extract> pairs by following the methodology described in Marcu (1999), Jing & McKeown (1999).

The methodology applied to learn summary content is one which follows a sentence classification paradigm where the system has to distinguish between two categories: *summary sentence* and *non-summary sentence*. This methodology has been proposed by Kupiec & *al.* (1995), and applied many times in the development of summarization systems.

The learner system is given the annotated examples, e.g., documents where some sentences have been marked as relevant and the rest is assumed to be non-relevant, in order to predict how to select sentences from previously unseen documents. Features such the ones introduced before are computed over all document sentences and used to create a model. In the following subsection we explain how this approach is implemented.

5.1. Corpora for Training Summarization Systems

There are a number of datasets available for text summarization experimentation and training of summarization components: The SUMMA conference (Mani & *al.* 2002) created a dataset which although does not contain human summaries, contains the summaries created by the system participants, and also a set of documents and questions to support question-answering based summarization (where the objective is to create a well formed summary answering a set of given questions).

The DUC conferences (Over & *al.* 2007) provides with a number of datasets for experimentation including single document summaries and multi-document summaries at multiple compression rates (400 words, 200 words, 100 words, and 50 words); extracts at different compression rates; topic-based summaries; biographies; headlines; etc.

The Ziff-Davis corpus is a collection of technical articles with human summaries. For each document a set of clauses from the text which can be considered close in content to the human summaries has been created (Marcu 1999). The extract for each document was created by an automatic program informed by corpus statistics. The CAST corpus (Orasan & *al*. 2003) is a collection of newswire texts and popular science articles where essential sentences have been identified by humans. These sentences may also contain fragments which are considered unessential for a summary.

The Summbank corpus is a resource for the study of monolingual, cross-lingual, multi-document and query-based summarization (when summaries are produced taking into account the interest of a user as expressed in a query) distributed by the Linguistic Data Consortium (LDC) under catalogue number LDC2003T16 and created as part of a John Hopkins University Workshop on Language and Speech (Saggion & *al.* 2002b ; Radev & *al.* 2003). The corpus is in fact an enriched version of the Hong Kong News Corpus also distributed by the LDC. A set of 40 document clusters where created around a set of topics or queries, where each cluster contains a set of ten documents considered relevant for the topic. Because the corpus is parallel, the clusters exist in English and Chinese. For each document, sentences have been marked and for each sentence, judgements about their relevance for a topic-based summary have been produced by three different assessors. These relevance judgements or utility values are integer numbers between 0 and 10 which reflect the relevance of the sentence to a summary (10 indicates very relevant and 0 indicates irrelevant). These values provide a fine-grained scale which can be used to produce a variety of gold standard extracts. Non-extractive summaries for each of the 40 clusters were also produced by human assessors at different word-based compression rates (50, 100, 200, and 400 words). A post-processed version of one of the documents in the corpus is shown in Figure 3 where sentences with relevance judgements (e.g., UTILITYn) are presented.

Figure 3. *Modified Document from the* SummBank Corpus *on the topic "Illegal Immigrants".*

The figure contains sentence mark-up (S) elements with features indicating what utility values have been given by each judge. This particular document is related to the topic "Illegal immigrants" (specified in the DOCSENT mark-up element) and the utility values have been assigned accordingly to the relevance of each sentence to that topic.

5.2. Experiments on Content Selection

The experiments described here consist on the application of a sentence classification program to decide whether or not a sentence belongs to a summary. In order to be able to carry out this type of experiments it is important that the data set to be used for training and testing has certain regularity and that during testing the documents are drawn from the same population, thus we have used subsets of the Summbank corpus which is a set of documents grouped by *human summarizer* (containing utility judgements for a given judged). Five such sets exist in the corpus. We assume that independently of the cluster (or type of query) the documents to be summarized belong to; the human summarizers will always follow a similar summarization strategy. The experiments reported here therefore try to simulate how a given human summarizer would select sentences from a document which is related to a query. Each subset comprises between 70 and 150 documents, a very reasonable training set. For a given summarizer J each sentence S was annotated as belonging to a summary if and only if the $utility_J(S) > threshold$ (e.g., the utility given by J to the sentence is greater than a threshold). Using this strategy and setting the threshold to utility 7 compression rates (in proportion of selected sentences) ranging from 14% to 60% on each cluster are obtained (see table below).

For each sentence we have computed the following features with the SUMMA toolkit: position feature, query similarity features, title similarity feature, term frequency feature. The documents were given to different learning algorithms including Naïve Bayes classification (Witten & Eibe 2005), k-Nearest Neighbours (Witten & Eibe 2005), and Support Vector

Machines (http://svmlight.joachims.org). The three classifiers perform similarly in our set, results for the SVM classifier are presented in Table 1. The results indicate average performance over two runs of the algorithm on held-out data (66% of documents for training, 33% of documents for testing).

	CASES		PRECISION			RECALL			F-SCORE		
JDG	Y	N	Y	N	ALL	Y	N	ALL	Y	N	ALL
JUD1	171	1,062	0.54	0.88	0.88	0.11	0.99	0.88	0.19	0.93	0.88
JUD2	1,320	1,886	0.89	0.85	0.86	0.73	0.94	0.86	0.80	0.89	0.86
JUD3	433	572	0.63	0.60	0.61	0.39	0.79	0.61	0.48	0.68	0.61
JUD4	1,271	814	0.62	0.60	0.62	0.81	0.37	0.62	0.70	0.45	0.62
JUD5	531	834	0.58	0.73	0.69	0.42	0.84	0.69	0.48	0.78	0.69

Table 1. *Classification for Summarization with Support Vector Machines*.

The table presents results per judge or human summarizer (first column), we indicate the number of summary-sentences and non-summary sentences in the judge document set; the compression in the whole set (percent of sentences in the set considered relevant for summary inclusion); and precision, recall and f-score metrics obtained by the classifier for each class (yes/no/all). F-score is taken as the harmonic mean between precision and recall when they are equally weighted. There is a great deal of variability between the performances which can be achieved in the different sets. In particular, high recall can not be obtained for judge number one in the "yes" category; this is in agreement with the fact that this judge in general gives lower utility values to sentences when compared with the other two judges assigning utilities to the same document.

6. Role of Semantic/Syntactic information

Various researchers have concentrated on the problem of generating good quality summaries by paying attention to the types of information in the source document which should be identified and brought into a summary. In concept-based abstracting (CBA) (Paice & Jones 1993; Oakes & Paice 2001) produce abstracts of technical articles in crop husbandry. Using extraction patterns, they identify in text semantic roles such as species, cultivar, high level property, low level property, etc., and generate abstracts using a fixed canned template which contain the expected types of information in the application domain. Teufel and Moens (1999) used rhetorical classification for scientific articles; they apply sentence classification to identify in the source document types of *rhetorical* information such as Background, Topic, Results, etc., they used a statistical approach borrowed from work by Kupiec & *al*. (1995). Saggion and Lapalme (2002) developed an information extraction type of approach to technical summarization, they used a rule-based system which extracted specific types of information in the document such as the Topic, Method, Results, Conclusions, etc. They have shown that their approach outperformed generic summarization in different evaluation frameworks including DUC 2002 (Farzindar & *al.* 2002).

These pieces of research have heavily depended on either human annotation of full documents or on hand-coded rules and heuristics for developing the summarisers. Creating a corpus annotated with relevant information types is not only expensive but somehow limited because of the low agreement between human annotators. We are investigating the use of

available resources found in the Internet or in abstracting databases or abstracting services (LISA, CSA, INSPEC, etc.) to palliate for the limitations of human annotated data. In our current research we are investigating the use of available and already *annotated* summaries to train a sentence classification system which can in turn be used to bootstrap a text summarization system by sentence classification.

In addition to the benefits that abstracting services offer to the scientific community, the abstracts they publish can be used for a number of natural language processing tasks ranging from text classification and term extraction to text summarization. One type of abstract we are interested in is the structured abstract. Structured abstracts (Hartley & *al.* 1996) have a paragraph structure each being characterised by the use of sub-headings signalling their information type: Objective, Method, Result, Conclusion, etc. They are used in medicine and seem to be more useful than standard abstracts in the search for information. Also related to this type is the Problem structured abstract which is produced for papers reporting the solution of a scientific problem and they are characterised by the following information: Document Problem, Problem Solution, Tests, Related Problems and Content Elements (Trawinski 1989). These abstracts can be thought as answering a number of research questions such as "What is the objective of the research?" "What are the methods applied?" "What results have been obtained?" "What are the main conclusions?"

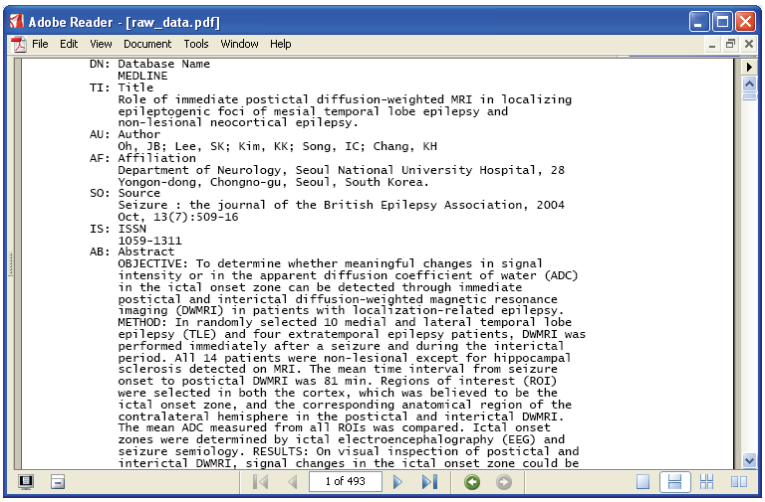

Figure 4. *Abstract from* Medline.

For this study we have collected around 1,500 abstracts from the Cambridge Scientific Abstracts service. We targeted the MEDLINE database searching for records containing explicitly the following keywords in their abstracts: "objective", "method", and "conclusions". This search yield records such as the one presented in Figure 4. We applied a number of scripts to the resulted set in order to transform the raw data into XML-structured representation which contains the following elements: a preamble section composed of the title, authors, source (journal), descriptors, etc.; and a text section containing the abstract which is composed of a number of informational categories such as objective, method, setting, results, etc., see Figure 5.

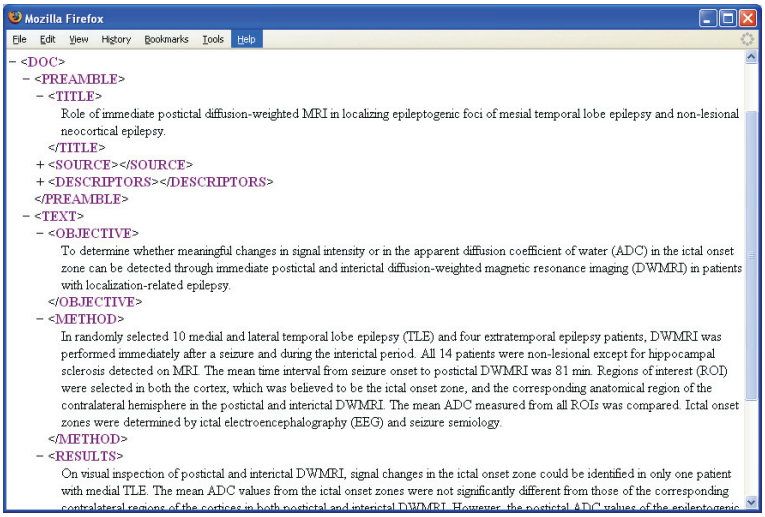

Figure 5. *XMLed Abstract.*

After creating this structure we discovered that there are various organisational or rhetorical structures present in the set of abstracts. One such structure is Background, Methods, Results, and Conclusions while other is Background and Objective, Patients and Method, Results, and Conclusions. Our objective is to use these abstract to automatically produce the rhetorical or informational structure of the abstract. Here, we only report results for the rhetorical structure: Objective, Methodology, Result, and Conclusion. We are also investigating other rhetorical structures and their relations to the one studied here, but experiments are still under way. We have used available tools to add linguistic information to the abstracts including tokenisation, sentence identification, parts of speech tagging, morphological analysis, and parsing and semantic representation using the SUPPLE parser (Gaizauskas & *al.* 2005). Each identified sentence is annotated with the rhetorical information of the segment where the sentence is found in the original abstract (e.g. sentences "In random selected…" and "All 14 patients…" in the OBJECTIVE section of the XML file presented in Figure 5 are annotated with the OBJECTIVE category). A number of features for each sentence in the abstract are also computed: these are two values for the position of the sentence in the abstract (position of the sentence from the beginning of the abstract, and the position of the sentence from the end of the abstract). We also indicate whether the sentence contains a word from the document title, record the length of the sentence in number of tokens, and identify if the sentence contains numeric information. Using the information from the parser we identify noun and verb phrases and their heads and extract a number of semantic triples such as subject-verb, object-verb, and noun-noun relations. These will be referred to as meta-level features. Machine learning components available in the GATE system are also used for the experiments. We have carried out sentence classification experiments using the collected corpus. We have followed the approach described by Li & *al.* (2004), who use SVM for sentence classification. In this approach each rhetorical category gives rise to a binary classification problem where a system is trained for each rhetorical category and the system has to decide if a sentence belongs to the target category. Two types of representation were used for the experiments:

(i) words were used as features (word features), and (ii) word features plus meta-information such as position, presence of title words, length (meta features). Cross-validation was carried out to measure classification accuracy.

System	Objective	Method	Results	Conclusions	Overall
W.Features	0.59	0.65	0.66	0.65	0.65
M.Features	0.89	0.75	0.84	0.74	0.79

Table 2. *Rhetorical Classification Experiments*.

The results per category and the overall results are presented in Table 2. Accuracy is reported in terms of F-score. A paired t-test shows statistically significant differences between the performance of the two systems; showing that meta-information helps the classifier.

While classification and rhetorical structure discovery is important in order to segment and make explicit the types of information in abstracts, it has also added benefits for full-document summarization. We are currently investigating a process of alignment and annotation transfer by which we first map the information of the summaries to full documents using monolingual alignment techniques (Jing & McKeown 1999) and then transfer the classification results to the full documents. The automatically annotated full documents can therefore be used for training full-document classification systems which rely on full-document sentence features.

7. Non-extractive Summarization

The generation of abstracts from textual sources has not been fully addressed by the research community, there have been however some studies on how to produce abstracts. In the Concept-based Abstracting method (Oakes & Paice 2001), a set of templates are used to express the information in the abstract. These templates contain canned text and variables which are filled in with information extracted from the source document. Depending on what variables have been extracted from the document different types of abstracts will be produced.

In cut-and-paste summarization (Jing & McKeown 2000), a set of hand-crafted rules have been developed by studying a corpus of text to summary transformations. These transformations include sentence reduction and sentence combination.

In the SumUM system (Saggion & Lapalme 2002) abstracts are generated by a process of text re-generation implementing transformation observed in a corpus of professionally written abstracts. For example, one transformation is to re-state information using the impersonal form of certain verbs (e.g. presentation verbs) so that a sentence such as “We present X” in the full document will appear as “Presents X” in the abstract; another transformation will eliminate parenthetical expressions, yet another transformation will present acronyms together with their explanation, etc. In spite of the multiple transformations taking place in human abstracting, only a few have been implemented in the SumUM system.

The best studied problem in non-extractive summarization is that of headline generation (Banko & *al.* 2000) where the objective is given a source document to produce a very short headline-like summary of the content. This problem has been addressed in a statistical framework where two sub-problems are modelled and combined: (i) the problem of what words or expressions to select from the document and (ii) the problem of how to create sentence out of a set of selected phrases. The system is usually trained using collections of

documents and headlines which are easily available. The problem of generating multi-sentence non-extractive summaries has not yet been fully studied.

8. Domain Specific Summarization

In spite of the advances in natural language processing over the past decades, it seems that most summarization solutions resists the incorporation of advanced linguistic techniques, however when summarization is carried out in a specific domain, linguistic analysis of some kind has proven to be beneficial. This has been the case in technical summarization and also in legal summarization. It is worth noting that in generic summarization linguistic information of a rhetorical type (Ono & *al.* 1994; Marcu 1997) has also been proved effective for selecting appropriate summarization content, however because of the cost associated with parsing full documents this is not generally adopted. Computation of lexical chains has usually showed a positive impact in extraction systems (Barzilay & Elhadad 1997).

In the context of legal summarization two types of linguistic (and textual) information are exploited. A text grammar which encodes knowledge about the organization of criminal cases is used in the SALOMON system to identify relevant parts of a document (Moens & *al.* 1997). Linguistic markers implemented as sequences of lexical items ("this court", "In reviewing section", "In conclusion", etc.) assists the identification of text segments in Farzindar and Lapalme (2004), showing that linguistic analysis is important in this domain.

In the context of technical summarization the SumUM system (Saggion & Lapalme 2002) produced indicative informative summaries in two steps. First a dictionary of generic scientific concepts and relations (e.g. problem, solution, conclude, introduce) and semantic patterns are used to locate useful information in source documents to produce an indicative summary. Noun phrases in the indicative abstract are presented to the user for informative expansion. Further patterns implemented to find definitions, statements of relevance and use, are used to locate sentences for an informative summary. Teufel and Moens (1999) used a classification system trained over annotated data and used it to identify types of information in scientific documents – a method refereed to as argumentative zoning. Paice and Jones (1993) implemented domain specific summarization by following an information extraction and template-based generation approach. Minel & *al.* (2000) system identifies specific types of information (definitions, thematic announcement, etc.) in documents in order to produce a summary which will take into account the particular types of information a reader is interested in. The information is identified by the use of linguistic markers and patterns specified by linguists.

9. Summarization in an Information Access System

The Cubreporter project at the University of Sheffield (Saggion & Gaizauskas 2006) uses summarization as one information access technology to allow journalist access a massive text collection of 11 years of news from the Press Association News Wire Archive. The system indexes articles using standard information retrieval technology and allows users to access information by formulating free and structured queries as well as well formed questions in natural language or which the system retrieves factoid type of answers. Generic single document summarization has been applied to the whole text collection to produce short summaries which are presented to the user in the results page. Sets of related stories (on the same news event) are also multi-document summarized using SUMMA, and access to the multi-document summaries allowed through the interface. Journalists can select and save to a basket a set of documents which are considered relevant and summarize the set of documents

using a centroid-based summarization technique. For each person appearing in the news (identified though named entity recognition software) a set of articles have been identified and person profiles have been created using linguistic patterns combined with statistical techniques (Saggion & Gaizauskas 2005). These profiles are stored in the system database and made accessible to the user through a user interface or whenever a use queries the named person (e.g. "Meeting between Tony Blair and George Bush"). The components of the system have been intrinsically evaluated with very positive results.

Other systems using text summarization technology for information access are NewsBlaster (McKeown & *al.* 2002) and NewsInEssence (Radev & *al.* 2005). The NewsBlaster system which collects information (news) from different news sites providing an interface to the news. Articles are assigned to news categories and clustered into events. Depending on the type of cluster (cluster about a single person, single event, etc.) different summarization strategies are applied. NewsInEssence at the University of Michigan searches dozens of news sites to cluster related stories and generates a summary highlighting the most important content. It uses the MEAD system and in particular a centroid-based method to select relevant sentences from a cluster.

10. Summarization Prospects

Text summarization is a key information access technology; it is interesting not only from the practical point of view in order to help users access and distil the ever increasing amount of information on-line but also from the theoretical point of view because in humans, summarization is associated to the processes of reading, understanding, and text production. It is also an interesting testbed for Artificial Intelligence theories of understanding, in the sense that one could use summarization techniques in order to test the cognitive capabilities of an artificial agent. From the linguistics point of view text summarization offers many challenges because studies on how to rephrase reduce and combine text fragments to produce a coherent piece of discourse are much needed in order to support/inform non-extractive summarization algorithms.

Text Summarization research is much alive and it will continue to attract the interest of researchers and practitioners in computational linguistics for many years to come. In spite of the evident gap between automatic summaries and human summaries and the need for further research in the field, summarization technology can now be seen in many commercial applications outside the research laboratory. This overview paper has presented a number of summarization concepts including classical approaches to selection of information from documents, evaluation and summarization in practical settings.

University of Sheffield, Department of Computer Science / Natural Language Processing Group
Regent Court
211 Portobello Street, Sheffield S1 4DP, England, UK
+44 114 222 1947 (tel/office) / +44 114 222 1810 (fax)
<h.saggion@dcs.shef.ac.uk>

References

Banko, M., Mittal, V.O., Witbrock, M.J. (2000). Headline generation based on statistical translation. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics.*

Barzilay, R. & Elhadad, M. (1997). Using Lexical Chains for Text Summarization. In *Proceedings of the ACL/EACL'97 Workshop on Intelligent Scalable Text Summarization.*

Baxendale, P.B. (1958). Man-made Index for Technical Literature, an experiment. *IBM J. Res. Dev.* 2(4).

Cremmins, E.T. (1982). *The Art of Abstracting*. Philadelphia, ISI Press.

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V. (2002). GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the ACL 2002*.

Demetriou, G., Skadina, I., Keskustalo H., Karlgren, J., Deksne, D. & *al*. (2004). Cross Lingual Document Retrieval, Categorisation and Navigation Based on Distributed Services. In *Proceedings of the Baltic HLT Conference 2004.*

Edmundson, H.P. (1969). New Methods in Automatic Extracting. *Journal of the Association for Computing Machinery,* 16(2).

Endres-Niggemeyer B., Hobbs J., Spärck Jones K. (1993). In *Workshop on Summarising Text for Intelligent Communication*. Dagstuhl, Germany.

Endres-Niggemeyer, B., Maier, E., Sigel, A. (1995). How to Implement a Naturalistic Model of Abstracting: Four Core Working Steps of an Expert Abstractor. *Information Processing and Management*, 31(5).

Farzindar, A. & Lapalme, G. (2004). LetSum, an automatic Legal Text Summarizing system. In T.F. Gordon (ed.), *Legal Knowledge and Information Systems, Jurix 2004: the Seventeenth Annual Conference*, Amsterdam, IOS Press, 11-18.

Farzindar, A., Saggion, H., Lapalme, G. (2002). Summaries with SumUM and its Expansion for Document Understanding Conference. In *Proceedings of DUC 2002*.

Firmin, T. & Chrzanowski, M.J. (1999). An Evaluation of Automatic Text Summarization Systems. In Mani, I. & Maybury, M.T. (eds).

Gaizauskas, R., Hepple, M., Saggion, H., Greenwood, M., Knight, K. (2005). SUPPLE: A Practical Parser for Natural Language Engineering Applications. In *Proceedings of IWPT'05*.

Harman, D. (2007). Text Summarization Special Issue. *Information Processing and Management,* 43(6).

Hartley, J., Sydes, M., Blurton, A. (1996). Obtaining Information Accurately and Quickly: Are Structured Abstracts More Efficient? *Journal of Information Science*, 22(5).

Jing, H. & McKeown, K. (1999). The Decomposition of Human-Written Summary Sentences. In *Proceedings of SIGIR 1999*.

Jing, H. & McKeown, K. (2000). Cut and Paste Summarization. In *Proceedings of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics.* Seattle.

Kupiec, J., Pedersen, J., Chen, F. (1995). A Trainable Document Summarizer. In *Proceedings of the 18th ACM-SIGIR*.

Lancaster, F.W. (2003). *Indexing and Abstracting in Theory and Practice*. 3rd ed. Champaign, University of Illinois, Graduate School of Library and Information Science.

Li, Y., Bontcheva, K., Cunningham, H. (2004). An SVM Based Learning Algorithm for Information Extraction. In *Sheffield Machine Learning Workshop*.

Lin, C. & Hovy, E. (1997). Identifying Topics by Position. In *Fifth Conference on Applied Natural Language Processing.*

Lin, C.Y. & Hovy, E. (2003). Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics. In *Proceedings of HLT-NAACL 2003.*

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

Mani, I. (2001). *Automatic Summarization*. Amsterdam, Benjamins.

Mani, I. & Maybury, M. (1999). *Advances in Automatic Text Summarization*. Cambridge, MIT Press.

Mani, I., & Maybury,. M. (1997). *Proceedings of the ACL/EACL'97 Workshop on Intelligent Scalable Text Summarization*.

Mani, I., Klein, G., House, D., Hirschman, L., Firmin, T., Sundheim, B. (2002). SUMMAC: a text summarization evaluation. *Natural Language Engineering*, 8(1).

Marcu, D. (1997). From Discourse Structures to Text Summaries. In *Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization*.

Marcu, D. (1999). The automatic construction of large-scale corpora for summarization research. In *Proceedings of SIGIR'99*.

McKeown, K.R., Barzilay, R., Evans, D., Hatzivassiloglou, V., Klavans, J.L. & *al*. (2002). Tracking and summarizing news on a daily basis with Columbia's Newsblaster. In *Proceedings of the Second international Conference on Human Language Technology Research* (San Diego), San Francisco, Morgan Kaufmann, 280-285.

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

Moens, M.F., Uyttendaele, C., Dumortier, J. (1997). Abstracting of Legal Cases: The SALOMON Experience. In *ICAIL* 1997, 114-122.

Nenkova, A. & Passonneau, R. (2004). Evaluating Content Selection in Summarization: The Pyramid Method. In *Proceedings of NAACL-HLT 2004*.

Oakes, M.P. & Paice, C.D. (2001). Term extraction for automatic abstracting. In Bourigault, D., Jacquemin, C., L'Homme, M.C. (eds), *Recent Advances in Computational Terminology,* Amsterdam*,* Benjamins.

Okumura, M., Fukusima, T., Nanba, H., Hirao, T. (2004). Text Summarization Challenge 2 Text summarization evaluation at NTCIR. *SIGIR Forum*, 38-1, 29-38.

Ono, K., Sumita, K., Miike, S. (1994). Abstract Generation Based on Rhetorical Structure Extraction. In *Proceedings of COLING 1994*.

Orasan, C., Mitkov, M., Hasler, L. (2003). CAST: a Computer-Aided Summarisation Tool. In *Proceedings of Research Notes Sessions of the 10th Conference of The European Chapter of the Association for Computational Linguistics (EACL2003)*, Budapest.

Over, P., Dang, H., Harman, D. (2007). DUC in context. *Information Processing & Management*. 43(6).

Paice, C. (1981). The Automatic Generation of Literary Abstracts: An Approach based on Identification of Self-indicating Phrases. In Norman, O.R., Robertson, S.E., van Rijsbergen, C.J., Williams, P.W. (eds), *Information Retrieval Research*, London, Butterworth.

Paice, C. (1990). Constructing Literature Abstracts by Computer: Technics and Prospects. *Information Processing and Management*, 26(1).

Paice, C. & Jones, P.A. (1993). The Identification of Important Concepts in Highly Structured Technical Papers. In *Proc. of the 16th ACM-SIGIR Conference*.

Radev, D. & Hovy, E. (1998). Intelligent Text Summarization. In *Papers from the 1998 AAAI Spring Symposium*, Technical Report SS-98-06.

Radev, D.R., Jing, H., Budzikowska, M. (2000). Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *ANLP/NAACL Workshop on Summarization*.

Radev, D., Hovy, E., McKeown, K. (2002). Text Summarization. *Computational Linguistics*. 28(4).

Radev, D., Otterbacher, J., Winkel, A., Blair-Goldensohn, S. (2005). NewsInEssence: summarizing online news topics. In *Communications of the ACM* 48(10).

Radev, D., Teufel, S., Saggion, H., Wai Lam, W., Blitzer, J. & *al*. (2003). Evaluation Challenges in Large-Scale Document Summarization. In *Proceedings of ACL 2003*.

Radev, D.,Allison, T., Blair-Goldensohn, S., Blitzer, J., Celebi, A. & *al*. (2004). MEAD - a platform for multidocument multilingual text summarization. In *Proceedings of LREC 2004*.

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

Salton. G., Singhal, A., Mitra, M., Buckley, C. (1997). Automatic Text Structuring and Summarization. *Information Processing and Management,* 33(2)*,* 193-207.

Saggion, H. (2002). Shallow-based Robust Summarization. *ATALA Workshop*. Paris.

Saggion, H. (2005). Topic-based Sumarization at DUC 2005. *Document Understanding Conference. HLT/EMNLP Conference*.

Saggion, H. (2006). Multilingual Multidocument Summarization: Tools and Evaluation. In *Proceedings of LREC 2006*.

Saggion, H. & Gaizauskas, R. (2004). Multi-document summarization by cluster/profile relevance and redundancy removal. In *Proceedings of the Document Understanding Conference 2004*, NIST.

Saggion, H. & Gaizauskas, R. (2005). Experiments on statistical and pattern-based biographical summarization. In *Proceedings of the Text Mining and Applications (TEMA) Workshop*.

Saggion, H. & Gaizauskas, R. (2006). Language Resources for Background Gathering. In *Proceedings of LREC 2006.*

Saggion, H. & Minel, J.L. (2005). Crossing Barriers in Text Summarization Research. *Workshop RANLP*, Bulgaria.

Saggion, H. & Lapalme, G. (2002). Generating Indicative-Informative Summaries with SumUM. *Computational Linguistics*, 28(4), 497–526.

Saggion, H., Radev, D., Teufel, S., Lam, W. (2002a). Meta-evaluation of Summaries in a Cross-lingual Environment using Content-based Metrics. In *Proceedings of COLING 2002.*

Saggion, H., Radev, D., Teufel, S., Wai, L., Strassel, S. (2002b). Developing Infrastructure for the Evaluation of Single and Multi-document Summarization Systems in a Cross-lingual Environment. In *Proceedings of LREC 2002*.

Spärck Jones, K. & Galliers, J.R. (1995). Evaluating Natural Language Processing Systems: An Analysis and Review. *Lecture Notes in Artificial Intelligence*, 1083.

Spärck Jones, K. (1999). Automatic summarising: factors and directions. In Mani, I. & Maybury, M.T. (eds), 1-12.

Spärck Jones, K. & Endres-Niggemeyer, B. (1995). Text Summarization. *Journal of Information Processing and Management*, 31(5), (Special Issue).

Teufel, S. & Moens, M. (1999). Argumentative classification of extracted sentences as a first step towards flexible abstracting. In Mani, I. & Maybury, M.T. (eds).

Trawinski, B. (1989). A Methodology for Writing Problem Structured Abstract. *Information Processing & Management*, 25(6).

Witten, I.H. & Eibe, F. (2005). *Data Mining*: *Practical Machine Learning Tools and Techniques*. San Francisco, Morgan Kaufmann.