An Automated Text Summarization Methodology

Shaidah Jusoh[a], Abdulsalam Masoud[b], Hejab M. Al Fawareh[c]
[a,b,c] *College of Arts and Sciences*
Universiti Utara Malaysia, 06010 Sintok, Kedah
Tel : 04-9284701, Fax : 04-9284753
E-mail : shaidah@uum.edu.my, a_sloma2003@yahoo.com, alfawareh@gmail.com

ABSTRACT

Most of the information is embedded in a long text documents. Having a summarizer that can produce a summary from the texts automatically is very desirable. This paper presents an introduction of an automated text summarization system by addressing the history of summarization and its existing application tools, and proposes a methodology for an automated text summarization. The proposed methodology utilized possibility and probability theory in the sentence extraction and sentence abstraction. The possibility and probability are also utilized in identifying relevant words and term occurrences techniques.

Keywords
Sentence Extraction, Sentence Abstraction, Possibility theory

1.0 INTRODUCTION

Automatic Text Summarization is a one of Natural Language Processing (NLP) fields, a subject area utilizing research in Linguistics, Computer Science, and Statistics. Researchers in this area always attempt to produce a program that is able to summarize textual documents in a way that a human does. According to Inderjeet Mani (year), "Text summarization is about the processing that extracts the most important information from a source to generate a short version of that source for a particular user or specific task". Other researchers have other definitions which define what text summarization essentially means, Radev (2003) defined summarization as "A short but precise representation of the document's contents", and "the automatic summarization main goal is taking information that comes on documents, extract its content, and present to the user its mainly important contents in a reduced form that satisfies the user's desire".

With extremely increasing available information and the limited time people have, retrieving information is a faster way is very desirable. The huge amount of all the available information that exists today is in a form of unstructured data, so called textual data. Textual information is embedded in textual data which is in a natural language forms; contained in books, product manuals, research papers, magazine articles, e-mails, and of course the Web. All of these would catalyzed the urgent needs of an automated text summarization tool. By having the tool, the textual information can be processed faster by human for decision making process. Therefore, the tool would be beneficial to all types of people in all kinds of domains. For example, in marketing department, the tool would help marketing managers or executive to identify the prospective markets in a short time. In human resource department, for example, the tool would help human resource managers to assign a job seeker to a right position in a shortest time. For public service departments, the tool can be used to extract customer complaints and so on. Although the tool that we dream of is difficult to achieve, the work to achieve the dream has been started since 50 years.

2.0 HISTORY OF SUMMARIZATION

The history of automatic computerized summarization has began 50 years ago . As the oldest publication, described an implementation of an automatic summarizer is often cited (Luhn, H. P. 1958). Luhn's method uses term frequencies to evaluate the acceptance of sentences for the summary. Its main idea is based neither on knowledge that significant words carrying most information are not too frequent nor too seldom in the text. Establishing boundaries of words significance by the help of their frequency would be a matter of experience. The consequence step is ranking the sentences, and reflecting the number of important words and the distance between those words in the sentence. At the end, it remains only to choose one or several highly ranked as a result.

The next significant progress was done ten years later. Edmundson (1969) introduced in his work hypothesis concerning high information value of title phrases, sentences that at the beginning and at the conclusion of articles usually, sentences that contain gesture words and phrases as (the results are, important, paper presents, the technique used, etc.). The renewed interest of the automatic summarization and the remarkable progress came in 90th, even if the next years brought more results (Gagnon & Sylva, 2005). With the growth of the Internet especially over the past few years, the exchange of information extremely has increased. However, the scientific groups make their scientific breakthroughs in state of awareness while on the other hand; reporters are presenting reports from all around the world that its information is updated in real time. That growing amount of books, magazines and electronic resources that are prepared

every day, puts huge pressure on those specialists as they struggle the overload with information.

3.0 SUMMARIZATION TECHNIQUES

Automatic text summarization is a technique that creates automatically a summary of a document or plain text. A text summarization system generates a summary of a text/document by extracting its most significant parts from the original text/document. In general, most of the summarization techniques are basically developed based on the manual text summarization approach that is to analyze natural language texts at the level of sentences individually. The main objective of this is to create and represents semantics of the important words and their relations in a sentence structure. The basic idea of automated summarization is to understand the whole meaning of the text which has been presented in form of a reasonably shorter text (summary). Figure 1 illustrates presents long texts as an input text and shorter texts as an output text. Summarization techniques are focused in the making of brief texts that can condense the content of a longer original text.

One of the summarization techniques is text categorization. Using the technique, the result of the summarization algorithm is a list of key-paragraphs, key-phrases or key-words that have been considered to be the most relevant ones. Although some methods are able to generate new sentences from the content, usually it consists in a pure selection of textual fragments.

Text Summarization is really a complex task itself, since a wide variety of techniques can be applied in order to condense content information, from pure statistical approaches to those using closer analysis of text structure involving linguistic and heuristic methods (anafora resolution, named entity recognition, lexical chains, etc,). In fact, many algorithms for feature reduction, feature transformation, feature weighting, etc. are directly related to this task, since they already try to select a proper and limited set of items that can be used as storing the core content of a given text. But the aim of summarization techniques is to go one step forward, rearranging this information to produce readable texts, although this processing is still in a very early stage (i.e. going from extraction to abstraction). Most of the working systems are based in the selection of a certain number of sentences found in the text which are considered to express most of the concepts present in the document.

Sentence extraction technique is usually statistical, linguistics, and heuristic methods, or a combination of all those techniques in order to generate a final summary. The result is not syntactically or content wise altered. Some of text summarizers such as *MEAD Summarizer* are based on this method (Radev, et. al, 2000).

Most existing Text Summarizers use the following framework:

1. Parse the document into sentences.
2. Determine the relevant sentences by ranking.
3. Use a subset of the ranked sentences to generate a summary (mostly the top ranked sentences).

Text summarization by extraction can be done in many ways, for example, paragraph, sentence, or keyword extractions.

Most of the summarization work done till the present date is based on extraction of sentences from the original text . The sentence extraction techniques compute score for each sentence based on features such as position of sentence in the document (Baxendale, 1958); , word or phrase frequency (Luhn, 1958), key phrases [terms that indicate the importance of the sentence in the summary e.g. “this paper focuses on”] .

Sentence-extraction text summarization systems use mostly statistic methods, linguistic and heuristic methods. Sentence extracts are done by identifying the most important sentences in the source document and combine them together so that produces what is likely to be a readable summary. The following steps are involved if one decide to use sentence extraction technique :

- Step 1: Divide the text into sentences.
- Step 2: Determine the sentence positions.
- Step 3: Determine the title words.
- Step 4: Determine Numerical values, Citations, or Bolded text.
- Step 5: Extract Named Entities. Frequency of the entities is used.
- Step 6: Extract keyword and determine its frequency.
- Step 7: Examine which word are present in the remaining sentences.
- Step 10: Give the rank of each sentence using combination function of all rankings with different weights.
- Step 11: Extract all the high ranked sentences.
- Step 12: Generate a summary.

Many existing methods determine which sentences in a document should be selected in the extraction process and some commercial systems using these methods such as Copernic Summarizer, Pertinence Summarizer and Microsoft Word’s summarizer .

Word-frequency-based rules method was initially introduced bt Luhn in 1959. The rules are used to identify sentences for summaries, based on the intuition that the most frequent words represent the most important concepts of the text. incorporated new features such as cue phrases, title/ heading words, and sentence location into the summarization process,

in addition to word frequency. The ideas behind these older approaches are still used in modern text extraction research.

MEAD Summarizer is a sentence-extractor which extracts essential sentences to the overall topic of a document. It attempts to reduce the redundancy of the summary by getting rid of sentences that above a similarity threshold parameter . Some other approaches use other methods for sentence extraction like Natural Language Processing (NLP), and machine learning techniques.

Summarization work is not focused only for English language. For example, has developed a technique based on a sentence weighting ordering approach, which applied on Japanese Newspaper Domain; it computes the weights of important sentences. Using this technique, a summary is obtained after computing weights of the important sentences by sort the remaining sentences after eliminate the sentence that sum of characters exceeds a restricted character amount.

In 2004, proposed a full-coverage approach summarizer that uses an algorithm to extract sentences from the document taking the concept repeatedly measuring the similarity of each sentence.

On the other hand, proposed Fuzzy-Rough aided sentence extraction for text summarization. Key sentences of a document are extracted by using Fuzzy-Rough sets that estimate the relevance of sentences. That method uses senses instead of raw words to reduce the problem that sentences of the same semantic meaning but written in synonyms are treated differently. Semantic clustering is also included, which used to avoid selecting redundant key sentences.

introduced a new measure that called Information Measure which to capture a sentence prior. This (Information Measure) measures the prior domain knowledge carried by the sentence.

Beside a sentence extraction technique, an abstraction technique is widely used to produce full sentences from some texts. Instead of extracting sentences from text, sentences are reduced , , , , or regenerated from the scratch to produce new sentences . Strategy of cut-and-paste was also applied . Like abstracting manually, the authors would have to acknowledge six editing operations: reducing the sentences; combine them; transform them syntactically; paraphrasing its lexical; generalize and specify; and re-ordering the sentences .

Summaries that been produced using the abstraction technique are more similar to the person summarization process rather than the sentence extraction technique. However, if huge amount of information needed to be summarized, in this case the extraction method is much more efficient. Extraction is a lot better towards irregularities of all input text. It is failure-proof and less language dependent .

4.0 EXISTING SUMMARIZATION TOOLS

Brevity Documen Summarizer was developed by Lextek International Company that specialist in full-text search technologies and generating document summaries. The summaries can be customized as the user wish. It can either highlight either the key sentences or words in a document Lextek claimed that Brevity is able to generate an accurate document summary, highlight the important sentences and words, and finding the most key parts in a document. That makes the users determine easily the content of documents (Lextek International, 2002).

Copernic Summarizer has been developed by Copernic Inc, the search software specialist. To generate document summaries, the copernic summarizer uses the statistical model (S-Model) and knowledge intensive processes which is (K-Process). It is able to generate a summary reports with key concepts and key sentences in four languages (English, Spanish, German, and French) in many forms (Documents, Hyperlinks, Website, Emails, and many other formats). Besides generating summaries, it also have the ability to highlight the concepts in the key sentences (Copernic Inc., 2009).

Extractor is a utility for web content summarization developed by the Interactive Information Technology Group at the National Research Council of Canada over seven years. It has functionalities such as define and extract the Keyphrase automatically, as well as keyphrases for metadata, indexing, highlighting, web log analysis, and interactive query refinement (Extractor Technology , 2008).

Inxight Summarizer has been developed by Inxight Federal Systems Company. It is one of the development kits that Inxight introduced to help the developers incorporate summarization into their intelligent solution search systems. It basically finds the key sentences based on the document. Its most powerful features are: extracting a very intelligent summary from a document in a really short time (seconds), providing a quick document previews so that can professionally help determining relevance without reading the entire documents, and it comes with a support of many languages (English, Arabic, French, German, Chinese, Dutch, Farsi, Finnish, Japanese, Spanish, etc.) on many supported platforms (Inxight Federal Systems, 2008).

Columbia University showed how the extraction technology of the information could be used to classify relevant term types, such as people, places, etc, and technical terms, which can be the focus for documents, despite the domain of the document. By using several features for instance frequency and term type, the system can identify “the foci” in the text and find relationships between them. So, the summary in this system is based on the sentences and clauses that cover the foci and their plausible relationships.

Sinope Summarizer (Carp, 2009) is a summarization tool that utilized an artificial intelligence technology and natural

language processing techniques. The system understands three languages (English, German, and Dutch) and it has five stages in order to generate the summary: (1) Analyze the web page; which helps to determine which text should be summarized. (2) Understand the page's text; *semantic analysis* technique is used in this stage to generate the *semantic structure.* (3) Determine the important elements; in this stage the system uses to analyze the *Semantic Structure* advanced mathematical techniques; that to determine which elements should be remained and all irrelevant elements will be eliminated. (4) Generate the summary; the system transforms back the *Semantic Structure* to its original. (5) Rebuild the web page; which the new page has just the summary text.

There are two challenging issues in automated text summarization. The first one is how to produce a shorter version of text without loosing any valuable information from the original texts. As previously discussed, current techniques for summarization are mainly based on sentence extraction. Although this technique, somehow, can produce a shorter text, it might happen that the sentence which is not selected to be extracted also contains valuable information. This occurs, because the current techniques do not process the semantic of the texts as a whole. Although Carp Technologies BV (2009) has started to address this issue by analyzing the sentence structure, it still lots of work should be done in this area. The second issue is how to evaluate the correctness and accuracy of the summary. If to produce a shorter version of text is difficult enough, then the job of evaluating the shorter text will be much more difficult.

5.0 PROPOSED METHODOLOGY

Although some works in automated text summarization have been conducted on using abstractive technique, the number of works is less than by using sentence extraction. The major issue of using sentence abstraction is the text processor has to understand the whole text and generate a summary as a human does. Although this technique can produce better summaries, however, this technique is very difficult to be implemented. On the other hand, sentence extraction is less difficult to be implemented, compared to sentence abstraction, but it is being conducted without understanding the context. In this paper, abstraction and sentence extraction techniques are combined. The steps involved in our proposed framework are as follows;

5.1 Sentence Segmentation

Sentence segmentation is the first step in automated text summarization. Normally, to parse a paragraph of text, a simple and limited way of dividing it into sentences would be to use '.' to obtain their ends. Extending this to '.', '!', and '?' would handle more cases correctly. However, while this is a reasonable list of punctuation characters that can end sentences, this technique does not recognize the punctuation characters that appear in the middle of sentences. For example, a sentence "The book cost Mr. Ali $30.65." has `.' in the two places in a sentence where it does not mean the end of a sentence. In this methodology **Split method** is proposed to be used. Using the Split method on this input will result in an array with three elements, when we really want an array with only two. We can do this by treating each of the characters '.', '!', '?' as potential rather than definite end-of-sentence markers. Scan through the input text, and each time it comes to one of these characters, it needs a way of deciding whether or not it marks the end of a sentence. A set of predicates related to the possible end-of-sentence positions is generated. Various features, relating to the characters before and after the possible end-of-sentence markers, are used to generate this set of predicates.

5.2 Tokenization

After a paragraph has been segmented into sentences, each sentence will be tokenized. Tokenization is a process of breaking down a sentence into a list of words. In this work, a lexicon that consists of a dynamic knowledge that helps in parsing by providing phrases and words information to the parse engine is used.

5.3 Part of Speech Tagging

Each token will be attached with its part of speech. For example, noun will be attached to the word "city". The challenging issues in assigning a part of speech to a word is, one word may have more than one part of speech. For example, the word "place" can belong to verb or noun. To resolve the problem, disambiguation rule are created and used. Examples of the rules are; if a word (**w**) at ***i*** position is a preposition, then ***w*** at ***i+1*** position is belong to **noun** and if a word (***w***) at ***i+1*** position is a preposition then ***w*** at ***i*** position is belong to **verb**

5.4 Keyword Identification

Keywords will be extracted from the token list by using a dynamic lexicon. The dynamic lexicon is a lexicon that updates its contents automatically by adding new keywords from the paragraph that are not existed before. The words that can be categorized as keywords include title' words, thematic words and emphasize words.

5.5 Relevant Term Identification

After a keyword has been identified, the remaining words in a sentence are defined as candidate words. Each candidate word will be ranked with a degree of relevancy to the identified keyword. In ranking the candidate words, fuzzy approach is

applied, where each word will be ranked based on human common sense. For example if a word "university" is taken as a keyword, the word "lecturer" is possible to be very much relevant to the university, where we can say 0.9. However, the word department might be ranked less than the "lecturer" word, let say 0.4. The assumption is made that not all an entity university has an entity department. Thus the candidate words that have high degree ranks are considered as relevant words and the sentence that contain these words are possible to be extracted.

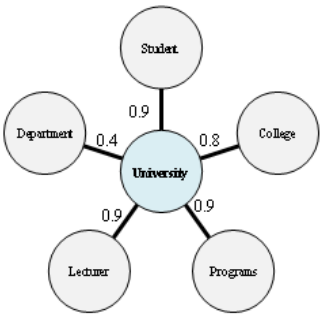

Figure 1: Example of a keyword and its candidate words.

5.6 Calculate Probability of Keyword and Relevant Term Occurrences in a Sentence

The probability theory is used to calculate the probability of keywords and it relevant words to occur in a sentence. Assume that "university" is a term to be considered as a keyword, and the relevant terms are "programs", "lecturer" and "students". The frequent terms can be obtained by counting term frequencies. The probability of keyword (***Pk***) term in a sentence is calculated as in the following equation

$$P_k = \frac{\sum k_s}{\sum k_d} \qquad (1)$$

where $\sum k_s$ represents the total frequencies of a keyword in a sentence and $\sum k_d$ is a total keyword in a document. The probability of relevant term (P_R) is calculated as

$$P_R = \frac{\sum R_s}{\sum R_d} \qquad (2)$$

where $\sum R_s$ represents the total frequencies of a relevant term in a sentence and $\sum R_d$ is the total of frequencies in a document. The probability of a sentence (***Ps***) can be formalized as

$$Ps = \sum P_{k(i-n)} \quad P_{R(i-n)} \qquad (3)$$

where is *i=1,* and *n= a finite number. Ps* of each sentence will be calculated and stored for further text processing usage.

5.7 Sentence Extraction

The sentence extraction is conducted on a sentence that has high probability value.

5.8 Sentence Refinement

Sentence refinement is conducted on the extracted sentences. In this stage, the understanding of the whole sentence will be conducted and the refinement is made by considering the context of the sentence. This step is conducted to ensure the selected sentences for a summary are precise and concise and unnecessary words are removed.

5.9 Summary Generation

The final step in automated text summarization is a summary generation. At this stage refined sentences are combined into a paragraph

6.0 SUMMARY

This paper presents a brief history of summarization and addresses the limitations of the existing summarization tools. Two well known techniques; sentence abstraction and extraction have been discussed. The challenging issues in automated text summarization have been highlighted as well. The paper also proposes a methodology for automated text summarization by utilizing sentence abstraction and extraction techniques. The methodology steps have been discussed.

ACKNOWLEDGEMENT

This research is supported by Ministry of Higher Education Malaysia under FRGS grant for a project entitled "Summarization of Malay Texts Document Using Artificial Intelligent"

REFERENCES

Carp Technologies BV. (2009). Sinope Summarizer. Retrieved Feb 2009, from English - Carp Technologies BV: http://www.sinope.info/en/index.php

Columbia University. (n.d.). FociSum. Retrieved Feb 2009, from Columbia University Text Summarization: http://www1.cs.columbia.edu/~hjing/sumDemo/FociSum/

Copernic Inc. (2009). Copernic Summarizer - Create concise document summaries. Retrieved Feb 2009, from Copernic - Software to Search, Find, and Manage Information: http://www.copernic.com/en/products/summarizer/