Text Summarization: Using Combinational Statistical and Linguistic Methods

Ghadeer Natshah
Information Technology
ppu
Hebron, Palestine

YasminTa'amra
Information Technology
ppu
Hebron, Palestine

Bara Amar
Information Technology
ppu
Hebron, Palestine

Manal Tamimi
Information Technology
ppu
Hebron, Palestine

Abstract— **This paper aims at proposing a methodology for automatic text summarization based on extractive methods. Extractive summarization needs the selection of a subset of important, meaningful sentences from the source text. The work presents a method for automatic text summarization of a single computer science article using a statistical and linguistic approach. About 50 experts' summaries were used to evaluate the proposed method. Our results show that using both the combinational statistical and linguistic methods is better than using each method alone. Results showed that 82% of the summaries produced by our approach satisfy the human summaries.**

Keywords- Text Mining, Extraction, Summarization, Statistical, Linguistic.

I. INTRODUCTION

Text mining is the process of retrieving information from different written resources to discovery of new information by understanding and analyzing a text, and finally finding important information hidden in it [1]. One of the text mining tasks is text summarization

Automatic text summarization is the technique where a computer summarizes a text[2]. A text is entered into the computer and a summarized text is returned, it will be a shorter version of the original text with preserving its information content without repetition [3].

The goal of automatic text summarization is reducing the original text into a shorter form containing it's information, main topics and overall meaning [4].

Text summarization systems are divided into extractive and abstractive methods that both aimed at analyzing the texts and generalizing summaries from them.

A. Extractive Summarization Method

This method works on selecting important paragraphs, sentences and key -phrases from the original document[5]. Then, it binds them into shorter forms. The extractive method consists of two approaches statistical and linguistic which decide the important sentence based on features of sentences[6]. Most of the recent automated text summarization system based on the extraction method because it is conceptually simple and easy to be implemented, an example of the extractive system is NeATS11 2001 [7].

B. Abstractive Summarization Method

This method is used to analyze and generate the summary through finding a way of understanding of the main concepts in a document, and then expressing those concepts in a clear, natural language by using linguistic knowledge. An example of the abstractive system is Cut & Paste2001 [8].

In this work, extraction approach is used to summarize a single English document. Specifically, in the domain of computer science articles.

.

II. RELATED WORKS

Previous studies have discussed a lot text summarization approaches. There are four different approaches for scoring and selecting key sentences as the following [9].

A. Statistical Methods

This extracts sentences that occurred in the source text, without taking into consideration the meaning of the words.

In [10], an extractive method for text summarization is suggested. This method is based on the statistical approach. Vishal used the idea of extracting the keywords, even if it is not existed explicitly within the text[11]. He achieved a design of the keyword extraction subsystem that helps in selecting the good sentences to be in the summary[12].

B. Linguistic Methods

This method needs to be aware of and known deeply in linguistic knowledge, so that the computer will be able to analyze the sentences and then decide which sentence is to be selected depending on the position of the subject, verb and name [13].

In [14], propose techniques for text summarization for a multi-document. They propose a new feature of the selection method to improve the summarization result using clustering methods. The similarity is calculated by a new formula that makes the cluster result more accurate[15].

C. Statistical and Linguistic Methods

It is a text summarization approach which uses both linguistic and statistical methods. This approach taking into account both linguistic and statistical hints to recognize terms[16].

In [17], this approach is used to summarize a domain specific text from the single web document. To do that, a used two novel features Sentence Weight and Subject Weight to rank sentences, and then they used a representative domain that is a specific corpus for the domain Direct Current (DC) electrical circuits. The Results showed that 68% of the resulted summaries satisfy the manual summaries.

D. Rhetorical Structure Theory (RST)

This theory is based on extraction the rhetorical structure (rhetorical relation) behind the decomposed text[18]. The rhetorical structure presents the logical connections between different parts of the text and the compound of the rhetorical relations between sentences. It can generate complete, correct and readable summarization on the basis of understanding the important sentence and the relation between them in original[19].

III. PROPOSED WORK

The proposed system is divided into four stages. The first stage is preprocessing the text, then the feature extraction which includes both word features and sentence features. The third stage is summary generation where the most important sentences are picked to generate the first summary. Finally, in post processing stage the final summary is generated with enhancement features. The architecture of our proposed system is shown in Figure (1).

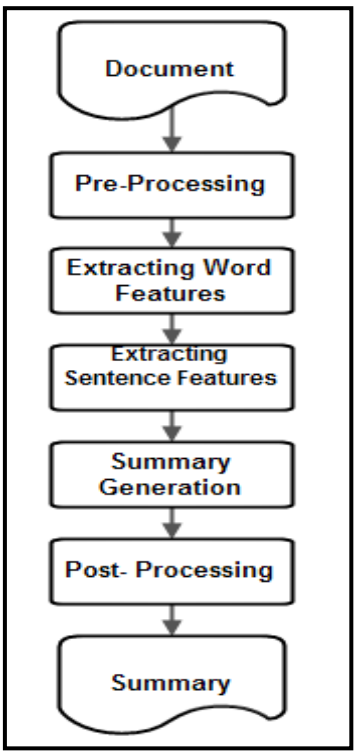

Figure1: Architecture of our proposed work

A. Preprocessing

The preprocessing is an important stage to prepare the input text for processing. This stage reduces the size of the input document, and removes the unimportant words. The preprocessing steps are:

a) Removing Stopwords: Stopwords are defined as general words that carry less important meanings than keywords. Such as " about, at, the… ". We removed the stop word, using list of 368 general words. We remove the stopwords by the following two steps. The first step classifies stopwords to either useless or useful words. Examples of useful words are (conclusion example, today…). But useless words are (at, the, some …), then Remove useless stopping words.

b) Referring abbreviations: We refer every abbreviation to its original word. The objective of this step is to know the real number of terms.

c) Segmentation: The document can be segmented into paragraphs if there are (.) and (\n). Then, every paragraph is separated into sentences by one of the three ends of sentence symbols: (.), (?) and (!).Those symbols can be used to split the text into individual sentences; there are some exceptions when one of those symbols does not indicate a sentence boundary. For example, MS., DR., and 100.00, the symbol (.) does not indicate sentence boundary. After that, every sentence is segmented into words depending on the spaces between the words. By the end of this step, we can get a word and mapping it with its address (paragraph, and sentence)[20]. This facilitates feature extraction because we can easily find the words and extract their features, also we can find the sentences and extract their features.

B. Extraction Feature

In the text summarization, the extraction of the features from words and sentences plays an important role. Each feature is given a value between '0' and ' 1 '.

a) Word Level Features: We basically focus on three features for each word. These features are*:*

- *Term Frequency- Inverse Sentence Frequency(TF_ISF):* Term Frequency: this is the number of times the term occurred in the document. Inverse Sentence Frequency: Terms that occur in only a few sentences are more valuable than ones that occur in many sentences. In other words, it is important to know in how many sentences of the document a certain word exists since a word which is common in a sentence, but also it is common in most sentences that it is less useful when it comes to differentiating that sentence from other sentences .ISF measures the information content of a word. The inverse sentence frequency is calculated with the following equation (1)

$$ISF_i = log\left(\frac{N}{n_i}\right) \quad (1)$$

The Term Frequency-Inverse Sentence Frequency is calculated by the following equation (2).

$$TF - ISF_i = TF \times log \quad (2)$$

Where N denotes the number of sentences in the document, and ni is the number of sentences in which term i occurs. TF-ISF feature is used as the following formula(3):

$$TF - ISF(w) = \begin{cases} 1, if\ TF - ISF \geq threshold \\ 0, if\ TF - ISF < threshold \end{cases} \quad (3)$$

- *Title Existence:* Titles and headings are considered as short summaries of the texts. The words that exist in the title or their synonyms are important and have high scores. This feature is computed as the following formula(4):

$$Existence\ in\ the\ title(\mathrm{w}) = \begin{cases} 1, if\ found\ in\ the\ title \\ 0, if\ not\ found\ in\ the\ title \end{cases} \quad (\mathbf{4})$$

- *Word Length*: Larger words occur less frequently than the smaller words, In order to negate this effect we considered the word length as a feature. This feature is computed as the following formula(5):

$$Length(w) = \begin{cases} 1, if\ length \geq threshold \\ 0, if\ length \geq threshold \end{cases} \quad (5)$$

b) Sentence level features: We basically focus on five features for each sentence. These features are.

- *Location of the sentence:* Location is computed as the following formula(6):

$$Location(s) = \begin{cases} 1, if\ the\ sentence\ location = threshold \\ 0\ , \quad Otherwise \end{cases} \quad (6)$$

- *Length of the Sentence:* This feature is computed as the following formula(8):

$$Length(s) = \begin{cases} 1, if\ length \geq threshold \\ 0, if\ length \geq threshold \end{cases} \quad (7)$$

- *Existence of Cue-Phrase:* Sentences may contain some cue phrase.

This feature is computed as the following formula(8):

$$Cue\ phrase(s) = \begin{cases} 1\ , if\ bonus\ phrase \\ 0\ , if\ Stigma\ phrase \end{cases} \quad (8)$$

- *Existence of keywords:* The weights for words are calculated as we mentioned earlier in this chapter. These weights are used to determine the keywords from the document's words. If the word's weight is more than such threshold is selected as a keyword is computed as the following formula(9):

$$KeyWord(w) = \begin{cases} 1, if\ weight(w) \geq threshold \\ 0\ , if\ weight(w) \leq threshold \end{cases} \quad (9)$$

The sentences which include keywords have more score, as formula(10) shows:

$$Existence\ of\ Keywords(s) = \begin{cases} 1, if\ number\ of\ keyword \geq threshold \\ 0, if\ number\ of\ keyword < threshold \end{cases} \quad (10)$$

- *Biased Word Feature:* If a word appearing in a sentence is from a biased words' list, then that sentence is important. Biased words' list is previously defined and may contain domain specific words. This feature is computed as formula(11):

$$Biased\ Word\ Feature(s) = \begin{cases} 1, if\ number\ of\ biased \geq threshold \\ 0, if\ number\ of\ biased < threshold \end{cases} \quad (\mathbf{11})$$

C. Summary Generation: Summary generation will be the third stage in this project. This stage will produce the first summary.

a) Sentence selection: at the beginning, we calculated the weight of every sentence by summing the scores. Then, we selected the sentences which have weight >= average of sentences weights to be in the first summary. After calculating every value for α, β, μ, Ω and φ , we calculated the weight for every sentences. The purpose of calculating the weight is to determine the most important sentences which is selected as a first summary. So, we did six experiments from which it is explored that the value of F-measure is approximated in the experiments 2, 3 and 4. Whereas, the values were respectively (0.60, 0.78, 0.79), and this will be explored by the figure (2).

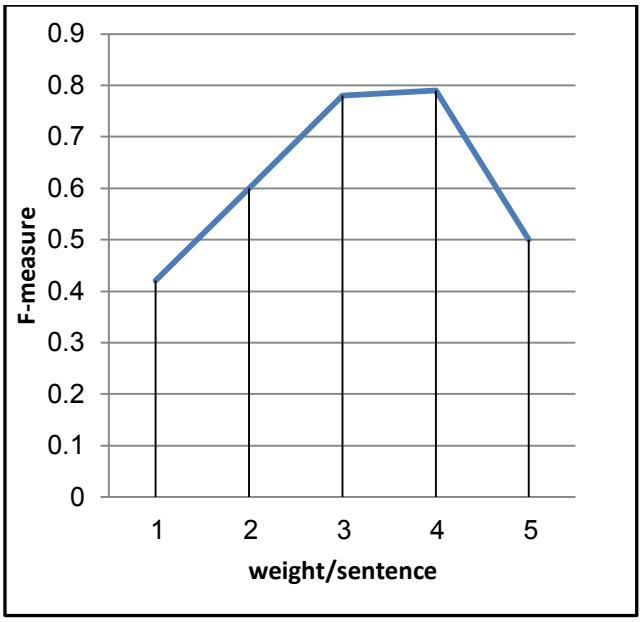

Figure 2: F-Measure values for sentences weights.

So that, we go to a seventh experiment which calculated the average of words weights deepening on the equitation (12).

$$Weight_{Average} = \sum_{s=1}^{n} \frac{weight(s)}{n} \quad (12)$$

Then, F-measure was calculated at the average for the words weights. Its value was 0.82 so it fulfilled the highest F-measure comparing with other experiments. The weight average was considered as a threshold. So the sentences which had weight more than this threshold were selected as first summary.

D. Postprocessing

After the extraction and generation of the first summary, some post processing steps will enhance the summary and produce the final summary.

a) Sentence ordering: Depending on the sentences' locations in the source, we ordered the final summary.

b) Connecting word: The appearance of connecting words such as "accordingly, again, also, besides… etc" at the beginning of the sentence provides coherence with the previous sentence[22]. Linguistically, a sentence cannot start with the above words without any related introduction in the previous sentence.

d) Removing URLs and E-mails: URLs and E-mails are not important details in the final summary. We used regular expressions in this step to select the URLs and e-mails then we removed them.

IV . Experiment And Result

The following contain the experiments and results for the project steps: preprocessing, features extraction, summary generation, and post processing.

A. Test Data Set

We used ten articles to test our work. These articles were chosen in a random way in the field of computer science. Then, we asked 50 experts to generate summaries. Each article was summarized by 50 experts, because the nature of the summary differs from one person to another, so, we did more than one summary for each article. Then we took these summaries and used them in evaluation process.

B. Evaluation Procedure

We had made different experiments to determine threshold to every feature. The purpose of determining threshold to every feature is to get the best summary. We used what are called Precision and Recall. This method is currently the most used method to evaluate extractive summarization.

Precision evaluates the proportion of correctness for the sentences in the summary whereas recall evaluates the proportion of relevant sentences included in summary. The weighted harmonic mean of precision and recall is called as F-measure:

$$Precision = \frac{Retrived\ Sentences\ \cap Relevant\ Sentences}{Retrieved} \quad (13)$$

$$Recall = \frac{Retrived\ Sentences\ \cap Relevant\ Sentences}{Relevant\ Sentences} \quad (14)$$

Relevant sentences = Sentences that are identified in the human generated summary

Retrieved sentences = Sentences that are retrieved by the system.

$$F - measure = \frac{2 \times Recall \times Precision}{(Recall + Precision)} \quad (15)$$

C. Results

Documents from the test set have been summarized using this system and manually by experts, and the selected sentences to be in the summary by each one is presented in figure (2) below:

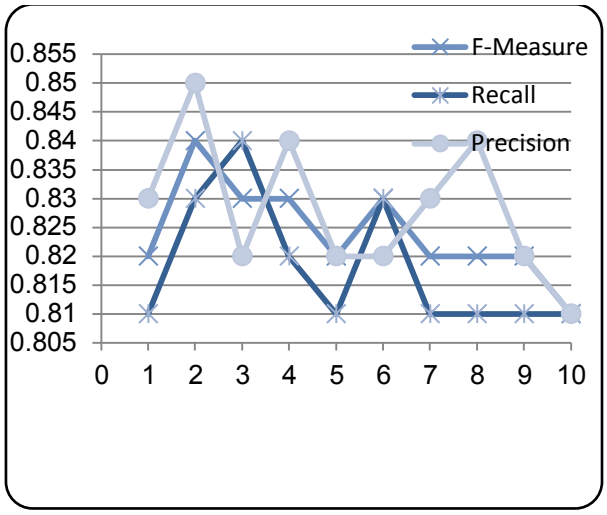

Figure 2:Different values of P,R,F-Measure for test data set summarized by the proposed system.

From this figure we noticed that, the values differ from document to another. These differences are depending on the nature that human summaries differ from one person to another.

V. Conclusion and Future Work

In this paper, new algorithms were developed to summarize domain-specific text from single document using linguistic and statistical methods. The development of these algorithms for automatic generation of text summarization allows us to contribute in an efficient way to the area of Natural Language Processing (NLP).

As a result of our paper, the combinational statistical and linguistic methods are better than each method alone. Results showed that 82% of the summaries produced by our approach satisfy the human summaries. This result became better than results using statistical or linguistic methods.

Extension to multi document summarization: Our proposed system summarizes single document summarization, and multi document is still a challenging extension of the current work.

Extension to generic domain summarization: Our proposed system summarizes documents in computer science field, and need to be extended to generic domain.

References

[1] Aliguliyev M ,(2009). A new sentence similarity measure and sentence based extractive technique for automatic text summarization. Expert Systems with Applications.

[2] Alonso A .(2005). Representing discourse for automatic text summarization via shallow NLP techniques. Master thesis. Department de Ling¨u´ıstica General. Universitat de Barcelona.

[3] Amini A& Gholamrezazadeh S.(2009) A Comprehensive Survey on Text Summarization Systems. IEEE. JOURNAL OF EMERGING TECHNOLOGIES IN WEB INTELLIGENCE, VOL. 1, NO. 1, .

[4] Arcelon d. (2008). What is automatic text summarization? Retrieved from: http://people.dsv.su.se/~hercules/textsammanfattningeng.html (accessed: Nov 09, 2010)

[5] Basu L.2009.Ruby WordNet. Retrieved from http://deveiate.org/projects/Ruby-WordNet (accessed: Apr 09, 2011).

[6] Battley L.2010. text . Retrieved from: http://rubygems.org/gems/text (accessed: May 01, 2011).

[7] Berkeley G.(2003). What Is Text Mining?. Retrieved from: http://people.ischool.berkeley.edu/~hearst/text-mining.html (accessed: Nov 19, 2010).

[8] Berube D.2007. Practical Ruby Gems. Printed and bound in the United States of America .

[9] Buss k.(2007). Literature review on preprocessing for text mining. Institute of creative technologies. Retrieved 41arcelon 25, 2009 from. http://www.ioct.dmu.ac.uk/.

[10] Chakrabarti P, Basu J.(2010). text summarization and discovery of frames and relationship from natural language text. (ijcse) international journal on computer science and engineering vol. 02, no.

[11] Chang T, Hsiao F w.(2008). A hybrid approach to automatic text summarization.IEEE.

[12] Chengcheng L.(2010).Automatic Text Summarization Based On Rhetorical Structure Theory. IEEE. 20I0 International Conference on Computer Application and System Modeling (ICCASM 2010).

[13] Collingbourne H,2009. THE BOOK OF RUBY . Retrieved from http://www.sapphiresteel.com/

[14] Dalianis S, (2005) .greeksum a greek text summarizer. Master thesis. George Pachantouris.

[15] Dongrui W, Jerry M. Mendel, Jhiin ." Linguistic Summarization Using IF-THEN Rules".2010 IEEE.

[16] Fattah A.(2008).Automatic Text Summarization.World Academy of Science, Engineering and Technology http://www.doc.ic.ac.uk/~lkhw98/project

[17] GitHub M.2011. Development Kit. Retrieved from: https://github.com/oneclick/rubyinstaller/wiki/development-kit (accessed: May 01, 2011)

[18] Jerry G.2010. NetBeans IDE 6.9.1 . Retrieved from http://netbeans.org/community/releases/69/

[19] Jezek K.(2007). Automatic Text Summarization.IEEE World Academy of Science, Engineering and Technology 2007

[20] LeiLI L, Ying X, Hongyan L.(2010). Multi-Document Summarization Based on Improved Features and Clustering. IEEE.

[21]Louie J. (2006). A hmm – viterbi algorithm based part of speech tagger. 3rd national natural language processing symposium building language tools and resources

[22] Manna S , b. Sumudu U.(2007). Fuzzy word similarity: a semantic approach using wordnet. IEEE.

[23] Mehdi B, Bashiri H.(2010). Evaluation of Clustering and Summarizing in Distributed Latent Semantic Indexing . IEEE.