Neural network based approach to study the effect of feature selection on document summarization

Dipti Y. Sakhare
Research scholar Bharati veedyapeeth deemed university, Pune,
Maharashtra, India

Dr.Rajkumar
DRDO Scientist 'D',
DIAT,Pune Maharashtra ,India
E-mail: diptiysakhare@gmail.com

Abstract

As the amount of textual Information increases, we experience a need for Automatic Text Summarizers. In Automatic summarization a text document or a larger corpus of multiple documents are reduced to a short set of words or paragraph that conveys the main meaning of the text. In this paper we proposed various features of Summary extraction. In the proposed approach during training phase, the feature vector is computed for a set of sentences using the feature extraction technique. After that, the feature vector and their corresponding feature scores are used to train the neural network optimally. Later in the testing phase, the input document is subjected to pre-processing and feature extraction techniques. Finally, by making use of sentence score, the most important sentences are extracted from the input document. The experimentation is performed with the DUC 2002 dataset. The features that are to be applied depending upon the size of the Document are also analyzed. The comparative results of the proposed approach and that of MS-Word are also presented here.

Keywords: Text Summarizers, features, extraction, pre-processing, DUC 2002 dataset

1. Introduction

Nowadays, enormous amount of digitally stored information is available on internet. In order to avoid this sinking, it is required to filter and extract information is necessary. An opportune tool which assists and interprets huge amount of text presented in documents is automatic text summarization (ATS).

The objective of ATS is to make a brief but adequate version of the original text retaining the most significant information [1]. The summary should meet the major concepts of the original document set, should be less redundant and sequenced. These are the basic attributes of the summary generation process. These attributes, along with the features selected, score the sentences to be included or rejected in the summary.

There are number of techniques of text summarization. Single document summarization creates the summary from a single text document. Multi-document summarization briefs a collection of related documents; in single summary. User-focused summaries process the text according to the initial search query; whereas generic summaries summarize the general perception of the document's content. Abstractive summary methods create the abstracts by interpreting the text using various linguistic methods. Extractive summarization methods do not interpret the text on the other hand they select best-scoring sentences from the original document based on a set of features selected.

The rest of the paper is organized as follows: Section 2 describes the review of recent works presented in the literature. Section 3 describes the pre-processing step. Section 4 presents the mathematical modelling for feature selection. Section 5 presents the results and discussion. Section 6 concludes the paper.

2. Literature survey

Automated text summarization is an old eminent research area and dates back to the 1950s. As a result of the information overloading on the web there is large-scale interest in automatic text summarization during these days.

The early work on single-document summarization was done by Luhn [3]. He presented a method of automatic abstracting in the year 1958. This algorithm scans the original text document for the most important information. The features used here are word frequency and sentence scoring. Depending on a threshold value for important factors the featured sentences are extracted. The Weakness of this system is the summary produced lacks in quality. The system was restricted too few specific domains of literature. Baxendale [4] useded sentence position as a feature to extract important parts of documents. Edmundson [5] proposed the concept of cue words. The strength of Edmundson's approach was the introduction to features like sentence position in text, cue words and title and heading words [5].

Pollock [6] Used sentence rejection algorithm. The aim of the paper was to develop a system which outputs a summary conforming to the standards of the Chemical Abstracts Service (CAS).

The abstractive summary generation was pioneered by ADAM Summarizer [7]. Machine Learning frame work is used to generate summaries using sentence ranking. The strength of this approach was it's potential to handle new domains in addition to redundancy elimination. K.R. Mc Keown in his thesis [7] generated the summary system using Natural Language Processing (NLP).The approach was based on a computational model of discourse analysis.

[11] Presented Term Weighting and Sentence Weighting as important features to recognize the featured sentences. It has also addressed the problem of anaphora resolution. Boguraev & Kennedy [10], Mercer [9] in 1997, Truney and Frank [8] in 1999, all of them used key phrases extraction as a supervised learning task. For these systems a separate training document set with already assigned key phrases is required to function properly. This is again an open challenge for research community.

Cut and Paste [12] is the first domain independent abstractive summarization tool. This was developed using sentence reduction and sentence combination techniques. Here a sentence extraction algorithm was implemented along with other features like lexical coherence, tf×idf score, cue phrases and sentence positions etc.

MEAD [13] was a multi document summarization toolkit it has used multiple position-based, TF×IDF, largest common subsequence, and keywords features. The methods for evaluating the quality of the summaries are both intrinsic (such as percent agreement, precision/recall, and relative utility) and extrinsic (document rank).A latest version of MEAD is based on centroid based multi document summarization.

[15] Has proposed keyword selection strategy. This is combined with the KFIDF measure to select the more meaningful sentences to be included in the summary. The Non-negative constraints used here are similar to the human cognition process. [14] Proposed a trainable summarizer based on feature selection and Support Vector Machine (SVM).Evolutionary connectionist model for ATS is developed by [16] which is based on evolutionary, fuzzy and connectionist techniques. All the papers discussed above use various features for summary generatin.Our aim in this paper is to perform the comparative study on the use of various features used for document summarization depending upon the size and type of the document. The following section describe the various steps in the proposed study.

3. Pre processing

The input document can be of any document format (doc, txt, rtf), hence the system first applies document converters to extract the text from the input document.

3.1 Text Prologuing

Pre-processing the text before incepting to summarization and categorization is Text Prologuing. It consists of three phases which are text segmentation, normalization and phase chunking.

3.1.1 Text Segmentation

This is the process of decomposing the given text into its constituent sentences, calculating each sentence length and word count. This module divides the document into sentences. At first glance, it may appear that using end of sentence punctuation marks, such as periods, question marks, and exclamation points, is sufficient for marking the sentence boundaries.

3.1.2 Normalization

is the process of converting words into normalized form. The following are the processes that come under normalization techniques.

3.1.3 Tokenization

It is the process of splitting of the sentence into words using String Tokenizer.

3.1.4 Stop word Removal

During the retrieval of relevant information we have to remove few words, numbers, and special symbols etc., which have less significance. A new approach is used for stop word removal. The stop words are classified as useful and useless stop word and the removed accordingly. This will help in faster operations at later stemming stage.

3.1.5 Case Folding

Converting entire words in the sentences into lower case so as to avoid repetition of same word in different cases like sentence case, capital case, title case, upper case etc.

3.1.6 Lemmatizing

Extracting the commonly featured, same meaning tokenized words so as to avoid repetition (e.g. problems-problem, risks-risk, etc.). It is a subset of stemming where only the suffixes are treated to clip or few entailments needed

3.1.7 Stemming

Mechanically removing or changing the suffixes of some nouns or verbs. Stemming improves the retrieval performance because they reduce variants of the same root word to a common concept. It also reduces the size of the indexing structure because the number of distinct index terms is reduced. The design of a stemmer is language specific, and requires some significant linguistic expertise in the language. Here we proposed an integrated stemming approach which involves both rule based approach and dictionary based approach. The proposed integrated model showed better impacting results with respect to words affected and computing time [17].

4. Mathematical modelling for feature selection

After pre-processing, the input document is subjected to feature extraction by which each sentence in the text document obtains a feature score based on its importance. The important text features used in the proposed system are: (1) Format based score (2) Numerical data (3) Term weight (4) Title feature (5) Co-relation among sentence (6) Co-relation among paragraph, (7) Concept-based feature and (8) Position data. The concept based feature is used for the first time.

4.1 Feature computation

4.1.1 Format based score: Expressing the text in diverse format E.g. Italics, Bold, underlined, big font size and more in many documents shows the importance of the sentences. This feature never depends on the whole document instead to some exact single sentence. Score can assigned to the sentence considering the format of the words in the text. The ratio of the number of words available in the sentence with special format to the total number of words in the sentence offers one to form the format which is dependent relative on the score of the sentence.

4.1.2 Numerical data: The importance stats concerning the vital purpose of the document are usually shown by the numerical data within the sentence and this has its own contributions on the basic thought of the document that usually make way to summary selection. The ratio of the number of numerical data that happens in sentence over the sentence length is thus used to calculate the score for this feature.

4.1.3 Term weight: Term weight is a feature value which is employed to look into the prominent sentences for summarizing the text documents. The term weight of a sentence is calculated as the ratio of the sentence weight to the maximum sentence weight in the given text document. The sentence weight is the summation of the weight factor of all the words in a sentence. The weight factor is the product of word frequency and the inverse of the sentence frequency.

$$TW = \frac{S_w}{\underset{i \in D}{Max}\left(S_w(i) \right)}$$

$$S_w = \sum_{j=1}^{n} W_j$$

$$W_i = TF \times ISF$$

$$ISF(t) = l\ og\ (\ N / N(\mathrm{T})\ )$$

Where, S_w → Sentence weight

W_j → Weight factor of the word in a sentence

n → Number of words in a sentence

TF → The number of occurrences of the term or word in a text document

ISF → Inverse Sentence Frequency

N → Total number of sentences in a document

$N(\mathrm{T})$ → Total number of sentences that contain the term (T)

4.1.4 Title features: A sentence is given a good score only when the given sentence has the title words. The intention of the document is shown via the word belonging to the title if available in that sentence. The ratio of the number of words in the sentence that occur in title to the total number of words in the title helps to calculate the score of a sentence for this feature.

4.1.5 Co-relation among sentence: At first, the correlation matrix C is generated in a size of NxM , in which N is the number of sentence and the M is the number of unique keywords in the document. Every element of the matrix is filled with zero or one, based on whether the corresponding keyword is presented or not. Then, the correlation of every vector with other vector (sentence with other sentence) is computed for all combinations so that the matrix of NxN is generated where every element is the correlation of two vector (two sentences). Then, every element of the row vector is added to get the sentence score.

4.1.6 Co-relation among paragraph: Here, the correlation is computed for every paragraph instead of sentences. for that, the correlation matrix C is generated in a size of PxM , in which P is the number of paragraph and the M is the number of unique keywords in the document. Every element of the matrix is filled with zero or one, based on whether the corresponding keyword is presented or not in the paragraph. Then, the correlation of every vector with other vector (paragraph with other paragraph) is computed for all combinations so that the matrix of PxP is generated where every element is the correlation of two vector (two paragraph). Then, every element of the row vector is added to get the score of every paragraphs and the score of every will obtain the same score of what its relevant paragraph obtained.

4.1.7 Concept-based feature: Initially, the concept is extracted from the input document using the mutual information and windowing process. A windowing process is carried out through the document, in which a virtual window of size 'k' is moved from left to right until the end of the document. Then, the following formulae is used to find the words that co-occurred together within each window.

$$MI(w_i, w_j) = \log 2 \frac{P(w_i, w_j)}{P(w_i) * P(w_j)}$$

Where, $P(w_i, w_j) \rightarrow$ The joint probability that both keyword appeared together in a text window
$P(w_i) \rightarrow$ The probability that a keyword w_i appears in a text window

The probability $P(w_i)$ is computed based on $\frac{|sw_t|}{|sw|}$, where sw_t is the number of sliding windows containing the keyword w_i and $|sw|$ is the total number of windows constructed from a text document. Similarly, $P(w_i, w_j)$ is the fraction of the number of windows containing both keywords out of the total number of windows. Then, for every concept extracted, the concept weight is computed based on the term weight procedure and the sentence score is also computed as per the procedure described in term weigh-based feature computation.

4.1.8 Position data: Position-based feature is computed with relevant to the sentence located in the document. With perspective of domain experts, initial sentence and the last sentence of the document is important than the other sentence. So, the maximum score is given for those sentences and the medium value is given to the sentence located in the starting and ending of every paragraph.

5. Feature matrix for training of feature-based neural network

This section describes the feature matrix used for training the feature-based neural network. The feature matrix is represented with the size of NxF , where N is the number of sentence and F is the number feature used in the proposed approach. (Here $F = 8$). Every element of the matrix is the feature score obtained for the corresponding sentence with the feature.

5.1 Training phase: Here, multi-layer perceptrons feed forward neural network is utilized for learning mechanism, in which the back-propagation algorithm is effectively utilized to train neural networks. To train the neural network effectively, the input layer is an individual (feature vector) obtained from the feature computation steps and the target output is zero or one that signifies whether its importance or not. Testing phase: In testing phase, the input text document is preprocessed and the feature score of every sentence in the document is computed. The computed feature score is applied to the trained network that returns the sentence score of every sentence presented in the input text document.

5.2 Ranking of sentence

Here, the ranking of sentence is carried out using the sentence score obtained from the previous step. Initially, sentences presented in the input text document are sorted in descending order according to the final sentence score. Then, the top-N sentences are selected for the summary based on the compression rate given by the input user. Finally, the selected top-N sentences are ordered in a sequential way based on the order of the reference number or unique ID to obtain the final summary.

$$N = \frac{C \times N_S}{100}$$

Where, N_S → Total number of sentences in the document

C → Compression rate

6. Results and Discussion

This section describes the detailed the experimental results and it and analysis of the document summarization. The proposed syntactic and sentence feature-based hybrid approach is implemented in MATLAB (Matlab7.11) and the experimentation is carried out with i5 processor having 3GM RAM.

6.1 DUC 2002 dataset

For experimentation, we have used DUC 2002 dataset that contains documents on different categories and extractive summary per document.

6.2 Experimental Results

At first, the input document is given to the proposed approach for document summarization. Then, the feature score is computed for every sentence based on the features utilized in the proposed hybrid approach. The sample results obtained for the feature matrix is given in table 1. This matrix is given to the neural network to obtain the sentence score. The final sentence score obtained from the neural network is given in table 2. Here, the neural network is trained with the sentences available in the DUC 2002(Cluster No. d071f and Document No. AP880310-0062) and the corresponding target label is identified with the summary given in DUC 2002 dataset.

Sentence ID	**Feature score**							
	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8
1	0	0	0.2500	0.4002	0.0695	0.1850	0.2307	0.2500
2	0	0	0	0.5695	-0.0044	0.1180	0.3283	0.2500
3	0.455	0	0	1.0000	-0.3568	-0.1640	0.5764	0.2500
4	0	0	0	0.3385	0.0141	-0.0790	0.1951	0
5	0	0	0	0.2733	0.2838	-0.0790	0.1575	0.2500
6	0	0	0	0.2470	0.6661	0.1386	0.1424	0
7	0.1000	0.1000	0	0.4426	0.0370	0.1386	0.2551	0.2500
8	0	0	0	0.5311	0.3792	0.4364	0.3062	0.2500

Table 1. Feature score for the text document (Cluster No. d071f and Document No. AP880310-0062)

Sentence ID	Neural network score
1	0.1518
2	0.1391
3	0.1648
4	0.0991
5	0.0752
6	0.0747
7	0.1164
8	0.1045

Table 2. Neural network score.

6.3 Performance Evaluation Measure

For performance evaluation, we have used the performance measure namely, precision, recall and F-measure. Precision measures the ratio of correctness for the sentences in the summary whereby recall is utilized to count the ratio of relevant sentences included in summary. For precision, the higher the values, the better the system is in excluding irrelevant sentences. On the other hand, the higher the recall values the more effective the system would be in retrieving the relevant sentences. The weighted harmonic mean of precision and recall is called as F-measure.

$$Precision = \frac{|\{\text{Retrieved sentences }\} \cap \{\text{Relevant sentences}\}|}{|\{\text{Retrieved sentences}\}|}$$

$$Recall = \frac{|\{\text{Retrived sentences}\} \cap \{\text{Relevant sentences}\}|}{|\{\text{Relevant sentences}\}|}$$

Where, Relevant sentences → Sentences that are identified in the human generated summary
Retrieved sentences → Sentences that are retrieved by the system

$$\text{F - measure} = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

6.4 Performance analysis

As per the application of above features, we analyzed that different types of documents requires different combinations of features to get precise Summary. We evaluated the summaries of different documents of Standard DUC Foundation. Documents are categorized.

Type 1 documents

Documents about a single short story not more than 15 sentences.

Type 2 documents

Documents about a biography of a person more than 15 sentences and less than 50sentences. Sentences.

We have compared MS Word Summary and our proposed approach using all eight features. The precision (figure 1), recall (figure 2) and f-measure (figure 3) for the two type of documents are evaluated.the results show that our proposed approach (using all eight features) outperforms the MS word summaries.

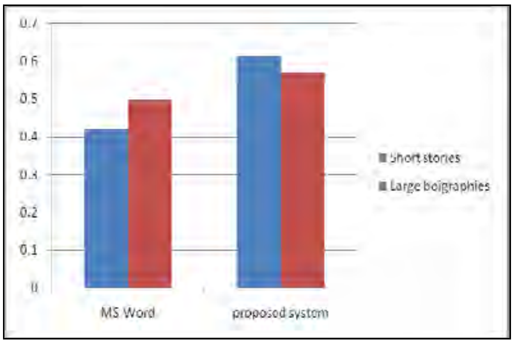

Figure 1.Effect on Precision

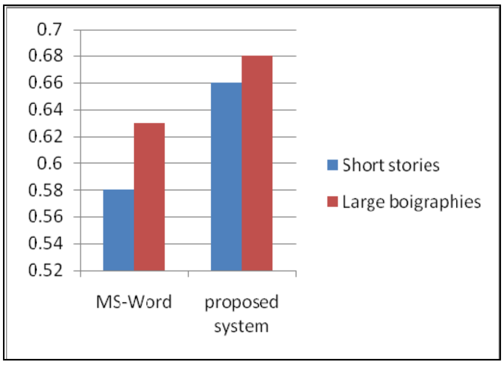

Figure 2.Effect on recall

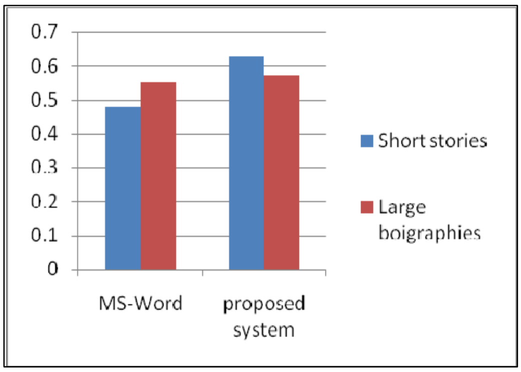

Figure 3. Effect on F measure

precision		
Document type	**Decision Module using all eight features**	**Decision Module using concept based features and set of any other 4 features**
Short stories	0.615	0.7
Biographies	0.47	0.47
Recall		
Document type	**Decision Module using all eight features**	**Decision Module using concept based features and set of any other 4 features**
Short stories	0.68	0.76
Biographies	0.6	0.5
F measure		
Document type	**Decision Module**	**Decision Module using concept based features and set of any other 4 features**
Short stories	0.68	0.75
Biographies	0.5	0.48

Table 3. Effect of concept based features on precision ,f measure and recall on short and large documents.

Precision, recall and f-measure values of summarization system using concept based features and a set of other 4 features are better as compared to summarization system without concept based for short stories. So applications which have to summarize short documents (typically containing 10-15 sentences) we can use concept based features. However, in case of large sized documents, it is essential to calculate all the eight features

7. Conclusion

As seen from the results, summary generated using proposed module outperforms to that of MS-WORD module for all the performance parameters. In future we will try to study the impact of individual feature on the summary generated.

8. References

[1] Automatic text summarization using sentence Features: a review, International J. of Engg. Research & Indu. Appls. (IJERIA). ISSN 0974-1518, Vol.4, No. IV ,November 2011, pp. 31 42

[2] Challenges and trends in automatic text summarization

[3] Luhn H.P, ―The Automatic Creation of Literature Abstracts, IBM Journal April 1958 pp. 159–165.

[4] Baxendale, P. (1958), ‗Machine-made Index for Technical Literature –An Experiment', IBM Journal of Research Development, Vol. 2, No.4, pp. 354-361.

[5] Edmundson H.P, ― New Methods in Automatic Extracting, Journal of the Association for Computing Machinery, Vol 16, No 2, April 1969, PP. 264-285.

[6] J.J.Pollock and A. Zamora , "Automatic Abstracting Research at Chemical Abstracts Service", Journal of Chemical Information and Computer Sciences, 15(4), 226-232(1975).

[7] Kathleen R. McKeown, ―Discourse Strategies for Generating Natural Language Text, Department of Computer Science, Columbia University, New York, 1982

[8] Turney. 1999. Learning to extract keyphrases from text. Teical chnReport ERB-1057. (NRC#41622), National Research Council, Institute for Information Technology.

[9] Marcu, D. 1999. The automatic construction of large-scale corpora for summarization research. In Proceedings of the 22nd International Conference on Research and Development in Information Retrieval, University of California, Berkeley, August.

[10] B K Boguraev, C Kennedy, R Bellamy, Dynamic presentation of phrasally-based document abstractions. 32nd International Conference on System Sciences, 1999.

[11] Brandow, R., Mitze, K., Rau, L. F. Automatic condensation of electronic publications by sentence selection. Information Processing anagement,31(5):675-685, 1995.

[12] Radev, R., Blair-goldensohn, S, Zhang, Z. Experiments in Single and Multi-Docuemtn Summarization using MEAD. In First Document Understanding Conference, New Orleans, LA, 2001.

[13] Jing, Hongyan and Kathleen McKeown. 2000. Cut and paste based text summarization. In 1st Conference of the North American Chapter of the Association for Computational Linguistics

[14] Nadira Begum, Mohamed Abdel Fattah, Fuji Ren, “Automatic text summarization using support vector machine”, International Journal of Innovative Computing, Volume 5, pp: 1987-1996, 2009.

[15] Rafeeq Al-Hashemi, “Text Summarization Extraction System (TSES)Using Extracted Keywords”, International Arab Journal of e-Technology, Vol. 1, No. 4, pp: 164-168, 2010

[16] Rajesh Shardanand Prasad, Uday Kulkarni, Implementation and Evaluation of Evolutionary Connectionist Approaches to Automated Text Summarization’ Journal of Computer Science 6(11):, 2010 ISSN 1549-3636, pp1366-1376.

[17] D.Y.Sakhare, Dr.Rakjumar ‘Syntactical Knowledge based Stemmer for Automatic Document Summarization’, CIIT international journal of data mining knowledge engineering print: ISSN 0974 – 9683 & online: issn 0974 – 9578 issue: march 2012 doi: dmke032012002.