Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 48 (2015) 728 – 734

Procedia Computer Science

International Conference on Intelligent Computing, Communication & Convergence

(ICCC-2015)

Conference Organized by Interscience Institute of Management and Technology,

Bhubaneswar, Odisha, India

Feature Priority Based Sentence Filtering Method for Extractive Automatic Text Summarization

Yogesh Kumar Meena[a,*], Dinesh Gopalani[b]

[a,b]*Malaviya National Institute of Technology, JLN Marg, Jaipur, 302017, India*

Abstract

Due to increasing amount of text data available on internet it becomes difficult for users to get the desired information quickly. In order to reduce this access time a summary could be utilized generated using Automatic Text Summarization. In general it could be extractive or abstractive. For extractive text summarization in which representative sentences from the document itself are selected as summary, various statistical, knowledge based and discourse based methods are proposed by researchers. In this paper we explored feature based extractive approaches for text summarization. We proposed a feature priority based filtering method for summarization. For this purpose we used sentence location as main feature and other features in priority to filter the redundant sentences. Experimental results on DUC2002 datasets shows that our method performs uniformly as compared to the best results for particular combination of features.

Peer-review under responsibility of scientific committee of International Conference on Computer, Communication and Convergence (ICCC 2015)

* Corresponding author. Tel.: +919461306647; fax: +01412529174
E-mail address: yogimnit@gmail.com

Peer-review under responsibility of scientific committee of International Conference on Computer, Communication and Convergence (ICCC 2015)
doi:10.1016/j.procs.2015.04.208

Keywords: Abstractive; Clustering; Statistical; Extractive; Summarization; Term Frequency;

1. Introduction

With the increasing amount of textual information over the internet it becomes difficult for the users to find the desired information quickly. They have to look up whole the document to get a glimpse of actual theme. Automatic text summarization solves the problem by generating summaries that could be utilized as a condensed replica of a document or a set of documents. Therefore Automatic Text Summarization can be defined as the process of condensing the source text document or set of text documents while retaining main information contents using a automatic machine. Automatic text summarization could be classified mainly as extractive automatic text summation and abstractive automatic text summarization. In Extractive automatic text summarization a subset of sentences from the original input set of sentences is selected as summary. In Abstractive automatic text summarization important topics in the textual unit are identified and new sentences are formed. Various Taxonomies are given from different aspects for text summarization by researchers. Spark Jones 1998 [1] Hovy & Lin 1998 [2] and Mani & Maybury 1999 [3] distinguished text summarization process from input, purpose and output factors. Research in the area of text summarization started from 1950's and till now no system is available that can generate summaries as like professionals or humans (Gold Summary). Most of the researchers focused on generic extractive summarization in their research contributions. Generic summaries basically represent overall information contents in condensed manner. The basic process flow of generic Extractive text summarization is shown in Fig. 1. Preprocessing is the first Step in which sentences are segmented using appropriate methods. In general „.' And „?' symbols are used as sentence end marker. Stopwords are removed as they do not convey much more information related to the actual topics of the text summarization. Stemming is performed using any standard method like Porter's Stemming Algorithm.

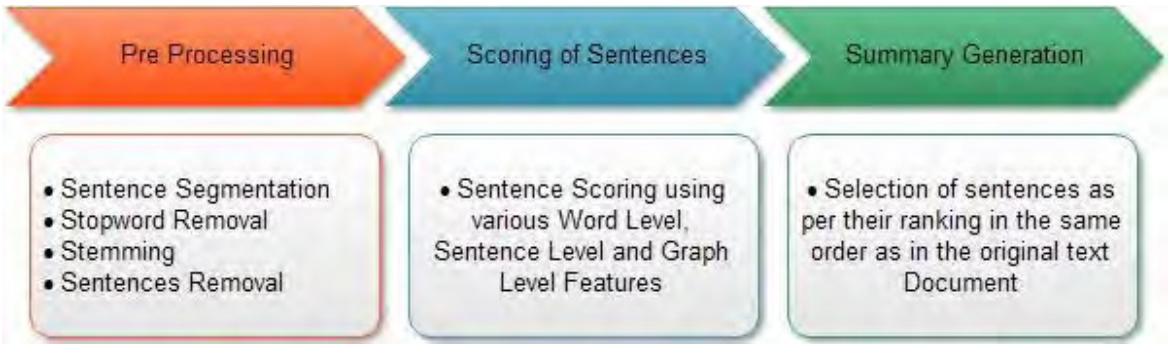

Fig. 1. Generic Extractive Summarization Process

The sentences containing unnecessary information for summary like "Figure" and "Table" are removed after it. After Preprocessing the sentences are scored using various word level, sentence level and graph level features. These sentences after scoring are selected in the same order as they appear in the original document in the summary generation phase. Other then generic summarization, update summaries, query focused summaries, sentimental summaries etc. can also be extracted for the text, but it depends on the purpose for which summarization needs to be performed. There are efforts made in abstractive automatic text summarization as well, but due the requirement of large domain knowledge and that too for specific domains is not preferred. Another reason is that information fusion that is the most difficult part of abstractive text summarization. Multi document summarization where for many documents a single summary needs to be generated is also important now a day due to different sources of the same topic of information. Statistical methods for extractive text summarization do not use much domain knowledge but instead they work on the available information content on the source document. Rest of the paper is organized as, in

section 2 we discuss related work in the area of extractive text summarization. In section 3 we discuss proposed feature priority based extractive text summarization method. Section 4 discusses results and performance evaluation. In section 5 we conclude the paper.

2. Related Work

In this section we will review work done so far in the area of extractive automatic text summarization. Research work started in this area started in 1950s but still we are lacking with efficient methods that can generate summaries like humans or professionals. Now a days with large volume of text content available online, much more efforts are required to perform text summarization. At the start Luhn [4] in 1958 proposed the method that utilized term frequency to score the sentences. He used sentence significance along with term frequency and summed up the frequency values for a particular sentence to get the score. Top scoring sentences were selected as the summary sentences. Later Baxendale [5] in 1969 proposed sentence location as a scoring criterion along with term frequency to score the sentences. Edmundsun [6] in 1969 proposed two more features for sentence scoring. Title similarity and Cue word feature also included for sentence scoring along with earlier two features. He proposed combinations of all four and analyzed the results for different combinations. Rush et al [7] in 1971 used sentence rejection methodology using different set of rules. Later it was used in 1975 for generation of chemical abstracts [15]. Later in 1997 Hovy and Lin [8] experimented that position method cannot produce efficient summaries for all domains. In 2001 MEAD [9] was developed that used features such as TF/IDF, Sentence Location, CueWords and Longest Common Subsequences for sentence scoring. Nobata et al 2001[14] used sentence location, sentence length, TF/IDF values of words, headline similarity and query to score significant sentences. They compared their method with TF-based and Lead-based methods. Varma et al 2005[12] used features such as sentence position, presence of verb in sentence, referencing pronouns, length of the sentence, term frequency, length of words, parts of speech tag, familiarity of the word, named entity tag, occurrence as heading or subheading and font style to score the sentences. Fattah and Ren 2009[11] proposed trainable models genetic algorithm, mathematical regression, feed forward neural networks, probabilistic neural network and Gaussian mixture model for text summarization. They used various features like sentence position, positive keyword, negative keyword, sentence centrality, sentence resemblance to the title, named entity in sentence, numerical data, sentence relative length, busy path of sentences and aggregate similarity for the purpose of training the model for text summarization. Prasad & Kulakarni 2010[16] used word similarity among paragraph, word similarity among sentence, iterative query score, format based score, numerical data, cue words, term frequency, thematic feature and tile similarity as features to score the sentences. The applied some evolutionary approaches as well for summary production. Abuobieda et al 2012[10] used title feature, sentence length, sentence position, numerical data and thematic words as features for scoring sentences. They used genetic algorithms to final design of the feature space. Rafael et al [17] in 2012 deeply analyzed all the sentence scoring features using ROUGUE [20] evaluation matrices. Mendoza et al 2014[13] used sentence position, title similarity, sentence length, cohesion and coverage as the features of the objective function. They used optimization techniques along with evolutionary algorithms for final summary generation. All above researchers scored sentences on the basis of some features. Most of the researchers gave equal weight to the all features included for sentence scoring. Rafael et al [18] in 2014 tried the combination s of different word, sentence and graph level algorithms for sentence scoring. In 2014 Meena and Gopalani [19] reviewed 22 features from available sources and analyzed the results on different combinations of features.

3. Proposed Work:

In our proposed approach we have tried to improve the efficiency as well as the readability of the summary. For the purpose of readability we have started our summary with the first sentence and ended with the last sentence of the summary, these sentences not only improve the readability but they are also part of the important sentences according to the sentence location feature. To select intermediate sentences we have used three features. The features that we have used are TF-ISF, Named Entity presence, Proper Noun presence. Term frequency of a term t in a document D is basically the number of times the term occurs in document D. To calculate Term Frequency we have used equation 1.

$$TF(t_i, d) = \frac{Frequency_i}{\max_j Frequency_j} \quad (1)$$

$$ISF(t_i) = \log_2\left(\frac{S}{s_i}\right) + 1 \quad (2)$$

Inverse sentence frequency is a variation of inverse document frequency, which suggests that if a term is less frequent in whole corpus than it is important for our document. Similarly ISF suggests that if a term is less frequent in whole document than it is more important for our sentence. Equation 2 is used to calculate ISF. Second feature that we have used is Named Entity presence. Stanford NER is used to identify all the named entities present in the sentence. The sentences are scored on the basis of Named Entities present in them. Lastly we have used Stanford POS Tagger for our third feature which is Proper Noun feature. According to this feature sentences which contain proper nouns are the most important and convey maximum information. These features are used to filter sentences at three levels. Proposed feature priority based sentence filtering algorithm is shown in Table 1.

Table 1. Feature Priority Filtering Algorithm

Algorithm 1: Feature Priority Filtering Algorithm
➢ Compute TF-ISF score for each term.
➢ Calculate TF-ISF Score of sentences on the basis of terms present in the sentence.
➢ Select top 50% sentences on the basis of TF-ISF score.
➢ Apply named entity recognizer on selected sentences.
➢ Select top 50% sentences on the basis of named entity presence.
➢ Apply POS tagging on selected sentences.
➢ Put the sentences in a list L in decreasing order of their score using proper nouns.
➢ To generate a summary of n documents select first sentence of the document and add it to the summary then select n-2 top sentences(other than first and last) from L add them to summary Then add the last sentence to the summary .

Our proposed approach consists of three steps.

Step-1 Preprocessing: In this step the data is preprocessed using the method that we have already discussed in this paper. In preprocessing we discarded the sentences whose length was less than 4 words or greater than 40 words.

Step-2 Sentence filtering: Sentence filtering has been applied in three levels. One feature is used to filter some sentences at each level. At first level we have used TF-ISF feature. The sentences are scored on the basis of TF-ISF score of the terms present in them. Equation 1 is used to calculate Term Frequency and equation 2 is used to Calculate Inverse Sentence Frequency of the terms. 50% of the sentences are filtered in this level. After this we score the sentences on the basis of presence of Named Entities in the sentence. Again 50% of remaining documents are filtered at this level. Finally on remaining 25% sentences we apply POS tagging to score the sentences on the basis of presence of Proper Nouns. After this the sentences are stored in decreasing order of their importance.

Step-3 Summary Generation: In this step we generate a summary with the help of top scoring sentences. The first sentence of the document is added to the summary first. After that we select n-2 top sentences from the list of sentences generated after POS scoring. These sentences are arranged according to their initial position in the document and added to summary. After these sentences the last sentence of the document is added to summary.

If sentences are scored as 0 using respective feature, where filtering has to be done they will be ranked as per the order they appear in the original text. If there is a tie in scores then limit of 50 percent can be extended till the same ranked sentences. As initially we are scoring the sentences using TF-ISF so there is not any problem. But

in case of POS tagging and Named Entity this issue might arise. While applying named entity, replacements of the same entity with different names should be considered as one entity.

4. Results & Analysis

We used 10 documents from DUC 2002 dataset to evaluate our algorithm. For the purpose of assessment of results we have used ROUGE-1 metric. ROUGE-q checks for the presence of each individual word in the system generated summary which is present in the gold summary. First we applied all possible combinations of 7 features namely Sentence location, Sentence centrality, TF-ISF, Sentence length cut-off, Named Entity recognition, Word co-occurrence, Proper noun presence. Then we applied our algorithm on all the 10 documents the relative performance is shown in Table 2 for F-Measure comparison, Table-3 for Recall comparison and Table 4 for Precision comparison.

Table 2: F-Measure Comparison

Document Number	Best Feature Set	F- Measure Value of Best Feature Set	F- Measure Value of Proposed Algorithm
1	4	0.53	0.41
2	1	0.41	0.36
3	2+4	0.34	0.39
4	3+5	0.42	0.43
5	1+3	0.45	0.46
6	2+5	0.68	0.64
7	5	0.43	0.39
8	3+6	0.46	0.44
9	4	0.30	0.28
10	2+4+5	0.67	0.60

1.TF-IDF, 2.Co-occurrence, 3.Sentence Centrality, 4.Sentence Location, 5.Named Entity, 6.Proper Noun

Table 3: Recall Comparison

Document Number	Best Feature Set	Recall Value of Best Feature Set	Recall Value of Proposed Algorithm
1	4	0.56	0.43
2	1	0.42	0.40
3	5	0.41	0.38
4	3	0.43	0.44
5	1	0.46	0.46
6	2+5	0.71	0.74
7	5	0.51	0.44
8	2+4	0.50	0.50
9	4	0.32	0.30
10	2+4+5	0.72	0.65

Table 4: Precision Comparison

Document Number	Best Feature Set	Precision Value of Best Feature Set	Precision Value of Proposed Algorithm
1	2	0.51	0.40
2	2+5	0.40	0.32
3	1	0.42	0.39
4	3	0.42	0.43
5	5	0.49	0.46
6	2+5	0.65	0.57
7	2+4	0.40	0.34

8	3+6	0.47	0.38
9	2+3+4+5	0.30	0.36
10	4+5	0.64	0.55

The results are not very high since ROUGE-N works on the basis of word to word matching and we have used abstractive gold summaries. In the table we can clearly see that most of the best outputs for each document came on different feature sets, if we concentrate on a single feature set, these results get worse. However our algorithm gave pretty consistent results and our output summary is also more readable than other approaches. In table 2, for document 3, 4 and 5 our results were better than the best feature sets. For document 2,6,7,8 and 9 also our results were comparable to best feature sets. However for document 1 and 10 they were pretty low, because in document 1Sentence location was more influential feature. However on an average our algorithm gave much better results than any other feature set individually. Also we can see in Table 3, for documents 2, 3, 4, 5, 6, 8 and 9 recalls are good and comparable to results of best feature sets. Table 4 is showing the comparison of precision of summaries. For most of the documents we got results as good as the with best feature sets.

5. Conclusion & Future Work

This paper suggests that it is not easy to find an efficient extractive summary of text using different feature combinations. Specific combinations can give higher efficiency but only on one or few documents. News documents especially our algorithm performs consistently on all documents as it takes the advantage of sentence location feature. TF/ISF, Named Entity and Proper Nouns are good indicators to include the sentences in the summary. Proposed approach may be extended with some semantic features with more filtering levels. Some discourse based features will also be included in future research.

References

1. K. S. Jones. Automatic summarising: Factors and directions. *In Advances in Automatic Text Summarization*, pages 1-12. MIT Press, 1998.
2. E. Hovy and C.-Y. Lin. Automated text summarization and the summarist system. In Proceedings of a Workshop on Held at Baltimore, *Association for Computational Linguistics.,* Maryland: October 13-15, 1998, TIPSTER '98, pages 197-214, Stroudsburg, PA, USA, 1998.
3. I. Mani and M. T. Maybury. Advances in Automatic Text Summarization. *MIT Press*, Cambridge, MA, USA, 1999.
4. H. P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159-165, Apr. 1958.
5. P. B. Baxendale. Machine-made index for technical literature: An experiment. *IBM J. Res. Dev.,* 2(4):354-361, Oct. 1958.
6. H. P. Edmundson. New methods in automatic extracting. J. ACM, 16(2):264-285, Apr. 1969.
7. J. E. Rush, R. Salvador, and A. Zamora. Automatic abstracting and indexing. ii. production of indicative abstracts by application of contextual inference and syntactic coherence criteria. *Journal of the American Society for Information Science*, 22(4):260-274, 1971.
8. Chin-Yew Lin and Eduard Hovy. 1997. Identifying topics by position. In *Proceedings of the fifth conference on Applied natural language processing* (ANLC '97). Association for Computational Linguistics, Stroudsburg, PA, USA, 283-290. 1997
9. Radev. D., Blair-Goldensohn S. And Zhang Z. (2001). Experiments in single and multi-document summarization using MEAD. In *First Document Understanding Conference*, New Orleans LA, 2001.
10. A. Abuobieda, N. Salim, A. Albaham, A. Osman, and Y. Kumar. Text summarization features selection method using pseudo genetic-based model. In International Conference on *Information Retrieval Knowledge Management (CAMP)*, 2012, pages 193-197, March 2012.
11. M. A. Fattah and F. Ren. Ga, mr, _nn, pnn and gmm based models for automatic text summarization. *Computer Speech and Language*, 23(1):126-144, 2009.
12. J. J, Pingali, and V. Varma. Sentence extraction based on single document summarization. In *Workshop on Document Summarization*, 2005.
13. M. Mendoza, S. Bonilla, C. Noguera, C. Cobos, and E. Leon. Extractive single-document summarization based on genetic operators and guided local search. *Expert Syst. Appl*., 41(9):4158-4169, July 2014.
14. C. Nobata, S. Sekine, M. Murata, K. Uchimoto, M. Utiyama, and H. Isahara. Sentence extraction system assembling multiple evidence. In Proceedings of the Second *NTCIR Workshop Meeting*, pages 5-213, 2001.
15. J. J. Pollock and A. Zamora. Automatic Abstracting Research at Chemical Abstracts Service. *Chemical Information and Computer Sciences*, 15(4):226-232, Nov. 1975.
16. P. R. Shardanand and U. Kulkarni. Implementation and evaluation of evolutionary connectionist approaches to automated text summarization. In *Journal of Computer Science*, pages 1366-1376, Feb 2010.

17. Rafael Ferreira, Luciano de Souza Cabral, Rafael Dueire Lins, Gabriel Pereira e Silva, Fred Freitas, George D.C. Cavalcanti, Rinaldo Lima, Steven J. Simske, Luciano Favaro, Assessing sentence scoring techniques for extractive text summarization, *Expert Systems with Applications*, Volume 40, Issue 14, 15 October 2013, Pages 5755-5764, ISSN 0957-4174, 2012.
18. Rafael Ferreira., Freitas F., De Souza Cabral L, Dueire Lins R., Lima R., Franca G., Simske S.J. and L. Favaro. "A Context Based Text Summarization System," 2014 *11th IAPR International Workshop on Document Analysis Systems (DAS)*, vol., no., pp.66,70, 7-10 April 2014.
19. Yogesh Kumar Meena and Dinesh Gopalani, Analysis of Sentence Scoring Methods for Extractive Automatic Text Summarization, *International Conference on Information and Communication Technology for Competitive Strategies (ICTCS-2014),* November 14 - 16 2014, Udaipur, Rajasthan, India , ACM 978-1-4503-3216-3/14/11, 2014.
20. Lin, Chin-Yew. 2004. ROUGE: a Package for Automatic Evaluation of Summaries. *In Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004*), Barcelona, Spain, July 25 - 26, 2004.