Text Summarization Algorithms: A Comparative Study

Yash Dhankhar, Rajiv Sharma
Computer Science and Engineering, Baba Mastnath Engineering College, Rohtak, Haryana, India
Yashdhankhar92@gmail.com, Rajivsharma02051984@gmail.com

Abstract— With the increasing amount of digital data, it has become difficult to retrieve the needed and concise information. Automatic text summarization caters to the very need of the time. It enables the reader to go through the essential contents in a brief period. Huge data available on the internet is required to be compressed so that the user can go through it and never miss the important set of information because of the enormous size of documents. In this paper, an analysis is presented on the Single document and Multi-document summarization algorithms on different domain datasets.

Keywords—*Text summarization, Automatic summarization, compression, single document summarization, multiple document summarization*

I. INTRODUCTION

Text summarization algorithms shorten the text and include only the vital information. There exist many online text summarizers which implement different algorithms to summarize the given text. This work aims at checking the accuracy of the present day text summarization algorithms. The algorithms used for analysis of single document summarizations are TextRank[5] (which further uses PageRank[14] to select sentences), Texteaser and summary tools based on the word features. The algorithms used for multiple document text summarizations include one based on Latent Dirichlet Allocation (LDA) [8] topic model to find out latent topics and topic distribution to select the sentences for final output summary. Other algorithm for Multi Document summarization which first selects the most important document in a set of documents using LexRank and then forms clusters for each sentence of important document aligning with the sentences of other documents sentences and finally finds a sentence from each cluster using Integer Linear Programming (ILP) method[11] with the aim to maximize the information content and Linguistic Quality Score.

II. SINGLE DOCUMENT SUMMARIZATION

Text summarization is done to shorten the text and get to the main point of the document. Summaries are easy to read and understand. Many single document summarization algorithms are available, few of them are analyzed for their efficiency in summarizing single documents.

A.Text Rank

Text Rank[5], an unsupervised algorithm based on weighted-graphs from a paper by Mihalcea et al. It is built on top of the popular Page Rank algorithm that Google used for ranking web pages. Text Rank works as follows:

1.Pre-process the text: remove stop words and stem the remaining words.
2.Create a graph where vertices are sentences.
3.Connect every sentence to every other sentence by an edge. The weight of the edge is how similar the two sentences are.
4.Run the PageRank algorithm on the graph.
5.Pick the vertices(sentences) with the highest PageRank score

In original Text Rank the weight of an edge between two sentences is the percentage of words appearing in both of them. This Text Rank uses a function to see how similar the sentences are. Graph based algorithms are used to rank the text sentences or words for summarization. To enable working with text on these algorithms, text is represented as graph, where a word depicts the nodes of the graph and edges represent meaningful relations among nodes. Edges represent the connection between two vertices of the graph. Sentences or collocations may also be assigned as vertices of the graph depending upon the size of input dataset. Edges may represent lexical relations, content overlap etc.
PageRank[14] presents a popular method to calculate the importance of a page in a set of pages joined together by links. It works by measuring the quantitative and qualitative score of links associated to a specific page. It computes an approximate score on the basis of that more websites are likely to contain forward links to important and popular websites. This algorithm analyse the links among different pages and assigns a numerical score to each element of the connected document set. It measures the relative importance of an entity in a set, like in the World Wide Web. The PageRank algorithm can be used to evaluate importance for any collection of elements which has references among themselves. For an element D, P(D) represents the associated PageRank.

B. Text Teaser

It is based upon sentence features[16], which is a heuristic approach for extractive text summarization.
Text Teaser associates a score with every sentence. This score is a linear combination of features extracted from that sentence. Features that Text Teaser looks at are:
• title Feature: The count of words which are common to the title of the document and sentence.
• sentence Length: Authors of Text Teaser defined a constant “ideal” (with value 20), which represents the ideal length of the summary, in terms of several words. Sentence Length is calculated as a normalized distance from this value.
• sentence Position: Normalized sentence number (position in the list of sentences). Introduction and conclusion will have a higher score for this feature.

• keyword Frequency: Term frequency in the bag-of-words model (after removing stop words). Keyword frequency is just the frequency of the words used in the whole text.

More on the sentence features for summarization see Sentence Extraction Based Single Document Summarization by Jagadeesh et al [16].

The process involved in Text teaser works in following sub processes:

1.Sentence Marker: It is used to split the document into sentence units.

2.Syntactic Parsing: It is done by sentence structure analysis using NLP tools like Brills tagger [Brill], named entity extractor, etc. This extractor recognizes named entities (like persons, organizations, and locations etc), temporal expressions (time and date) and specific numerical values expression from textual data.

3.Feature Extraction: Both the word level features are extracted to be used in the calculation of the relevance and importance of the sentence present in the document.

4. Sentence Ranking and Summary Generation: Most of the times word features depends on the context of its occurrence, i.e. they may depend on the sentence position and number also (ex. POS tag, familiarity, ..). Similarly, the word score also depends on the sentence number in the document. Once the feature vector is extracted for each sentence, the score of a sentence is calculated by obtaining the total sum of individual words as:

$$\text{Score (l , w)} = \prod_i fi(w)$$

$$\text{Score (l)} = \sum_i \text{Score (l ,wi)}$$

where l, represents the sentence number and ‘w’ represents the word present in the sentence, and f i (w) represents the ith feature value.

After the sentence scores are assigned, sentences are selected to form good summary. One method is to extract the top N sentences but this may lead to the coherence problem.

Coherence Score (CS): Coherence score [32] is used to identify the amount of common information between the set of already selected sentences and the new sentence to be included. A list of words is used to evaluate the coherence of the sentences.

Let Sw represents the set of words in the already selected sentences, and lw denotes the set of words present in the new sentence to be selected, then coherence score is obtained by the total sum of the common word scores. Now the score of the new sentence is computed by

$$\text{CF} \times \text{CS (l)} + (1-\text{CF}) \times \text{SPW (l)}$$

where CF denotes the Coherence Factor.

C. Summary Algorithm based on Word Features

This Algorithm [33] aims to provide an efficient manner of reducing a document to an understandable text, which is done by selecting the most important sentences.The core algorithm has 7 key steps listed below:

1. Associate each word with the grammatical equivalents. (e.g. "light" and "lights")
2. Compute the frequency of each word in the document.
3. Assign each word with points depending on their popularity.
4. Determine the correct ending of a sentence. (e.g "4.5" does not).
5. Separate individual sentences from the text.
6. Rank sentences based on obtained sum of associated words' points.
7. Select X topmost sentences.

III. MULTIPLE DOCUMENT SUMMARIZATION

Multi-document summarization is an automatic procedure to create a summary which includes important information on key topics from multiple documents. It creates a concise and comprehensive summary. Here, Algorithms are presented to perform summarization based on different methods to evaluate the accuracy of produced summary for different Multi-document datasets.

A. Multi-Document Abstractive Summarization Using ILP Based Multi-Sentence Compression

This Algorithm performs Multiple document summarization using integer linear programming model[34] which aims to produce coherent and highly informative sentences. First, Algorithm employs LexRank[35] to find out the most important document from the set of source documents. Then, the sentences belonging to the most important document are aligned to the sentences of another document to generate clusters of similar sentences. In each of the generated cluster, k-shortest paths from the sentences are generated with the help of word-graph structure. Finally, sentences are selected by the help of shortest paths generated employing a novel integer linear programming method in order to form new informative coherent sentences. Above stated shortest paths are represented as binary variables in the ILP method and number of words in a sentence path, information and quality score are considered in the function.

LexRank [35] creates a sentence graph where the edges represent weights which are calculated by the help of inter-sentence cosine similarities. While in this algorithm, a graph of documents is constructed to calculate the importance of a document. The equation below shows a formula to calculate LexRank score for a node in a graph using weighted links present among nodes. This computed score represents the importance of the document in the set of input documents. Let p(x) denotes the centrality of node x in the equation below:

$$\text{p(x)}= \frac{d}{n} + (1 - d) \sum_{v \in adj[x]} \frac{\text{idf}-\text{modified}-\text{cosine(u,v)}}{\sum_{z \in adj[v]} \text{idf}-\text{modified}-\text{cosine(z,v)}} p(v)$$

where adj[x] denotes the set of adjacent nodes to u and N represents the total number of nodes present in the graph, ‘d’ denotes damping factor(set to 0.85). Document representing the node with the highest LexRank score is a most important document, D_{imp} for the set of input documents.

B. Multi Document Summarization Algorithm based on LDA Topic Model

This Multi Document Summarization Algorithm[36] is based on the Latent Dirichlet Allocation (LDA) topic model which takes a multiple numbers of documents as input and generates a final output summary including an important piece of information from all the input documents. Latent Dirichlet allocation is a popular topic model which finds topics on the basis of word frequency i.e. occurrences of a word from a set of input documents. It presents the input text as a mixture of latent topics; these topics represent the key concepts in the document. LDA is particularly designed for identifying a reasonably accurate number of topics within a given document set. LDA (Latent Dirichlet Allocation) Model is used to find the important topics in the input provided.

These latent topics are useful to employ sentence ranking methods in order to obtain good quality summary. The sentence ranking mechanism calculates the posterior probability of each sentence based on two factors i.e. the topic distribution of the sentence and topic importance. Here, Topic Distribution denotes the degree to which a sentence belongs to a identified topic and Topic Importance denotes the importance of the topic depending upon the amount of information covered by this topic in the documents provided. After obtaining the probability for each of the existing sentences, it extracts the important sentences to be included in the final optimized summary based upon the above calculated posterior probability.

C. *Multi Document Summarization Algorithm using sentence clustering*

This Multi-Document text Summarization algorithm[37] uses clustering technique to extract an important piece of information from input documents. The sentence is considered as the most basic entity while performing Text Summarization. Clustering of sentences, paragraphs or text documents are performed on input dataset to produce a good multi-document summary.

This technique aims to produce Multi-document summary based on Single Document Summarization and sentence Clustering. In the algorithm, Single document summaries are produced by pre-processing and feature extraction of each document present in the dataset. The prepared summaries are combined by semantic based sentence clustering. Important sentences to be selected for final multi-document summary are chosen from these clusters with similar sentences. Non-redundant, coherent and important sentences are extracted for the summary.

IV. RESULTS AND ANALYSIS

A. *Datasets Used*

There are 3 datasets used to analyze the performance of all the algorithms under review in which dataset 1 includes the text files containing the data from newspapers, internet and news blogs regarding the news demonetization which contain 417 sentences and 40362 characters with spaces. Dataset 2 is a medical related data about a disease called Alzheimer's. The dataset includes the definition and introduction to the problem. Then it analyses the causes associated which are likely to cause Alzheimer's. It also includes the cure and how to approach the disease in the first phase which contain 356 sentences and 37045 characters with spaces. Dataset 3 includes cricket related data. It includes the history of cricket in India how it started and various milestones achieved in the times which contain 234 sentences and 29302 characters.

B. *Metrics Used Results and Analysis*

1. Similarity Score is a measure used for checking similarity among text data. It considers the familiar words and the position of words between system generated summary and human prepared summary. It returns the similarity score value in the range of 0 to 1.

2. ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is a recall-based measure which encourages the algorithms and systems to consider all the key topics in the summary. Recall measure can be calculated by using unigrams, bigrams or trigrams matching. For example, ROUGE-1 is evaluated as a count of unigrams in the system generated summary and reference summary.

3. BLEU metric can be described as a modified form of precision, generally used for machine translation evaluation. Precision represents the ratio of the number of familiar words in both gold and model translation/summary to that present in the model summary. Unlike ROUGE, BLEU takes the weighted average and directly accounts for variable length phrases.

C. *Results and Analysis*

1) *Single Document Summarization*

Table 1 describes the evaluated scores calculated by the Single document summaries prepared by algorithms: Text Rank, Text teaser and Summary by word features for News blog - Demonetization dataset i.e. Dataset 1.

Algorithm	**Similarity Score**	**ROUGE-1**	**Bleu metric**
Text Rank	0.72	0.576	0.269
Text teaser	0.58	0.473	0.311
Summary tool	0.52	0.475	0.206

Table 2 describes the evaluated scores calculated by the Single document summaries prepared by algorithms: Text Rank, Text teaser and Summary by word features for Medical-Alzheimer’s dataset i.e. dataset 2.

Algorithm	**Similarity Score**	**ROUGE-1**	**Bleu metric**
Text Rank	0.298	0.263	0.197
Text teaser	0.411	0.357	0.251
Summary tool	0.493	0.417	0.323

Table 3 describes the evaluated scores calculated by the Single document summaries prepared by algorithms: Text Rank, Text teaser and Summary by word features for Cricket related dataset i.e. dataset 3.

Algorithm	**Similarity Score**	**ROUGE-1**	**Bleu metric**
Text Rank	0.51	0.436	0.255
Text teaser	0.39	0.298	0.134
Summary tool	0.39	0.264	0.211

From these tables, we have analysed that the cricket domain data is best summarized by the Text Rank Algorithm. For medical dataset, Summary based on word features gives similarity score of 0.51, best among all other algorithms. For News related dataset, text Rank gives 0.72 similarity score

2) Multiple Document Summarization

Table 4 describes the evaluated scores calculated by the Single document summaries prepared by algorithms: Multi-document Summarization using ILP based method, LDA topic model and Summarization based on sentence clustering for News blog- Demonetization Dataset i.e. dataset 1

Algorithm	**Similarity Score**	**ROUGE-1**	**Bleu metric**
ILP based Sentence Fusion	0.24	0.29	0.18
LDA topic Model	0.38	0.431	0.34
Sentence Clustering	0.402	0.41	0.14

Table 5 describes the evaluated scores calculated by the Single document summaries prepared by algorithms: Multi-document Summarization using ILP based method, LDA topic model and Summarization based on sentence clustering for Medical- Alzheimer's Dataset i.e. dataset 2

Algorithm	**Similarity Score**	**ROUGE-1**	**Bleu metric**
ILP based Sentence Fusion	0.21	0.23	0.17
LDA topic Model	0.27	0.35	0.19
Sentence Clustering	0.43	0.34	0.28

Table 6 describes the evaluated scores calculated by the Single document summaries prepared by algorithms: Multi-document Summarization using ILP based method, LDA topic model and Summarization based on sentence clustering for Cricket Related Dataset I.e. Dataset 3.

Algorithm	**Similarity Score**	**ROUGE-1**	**Bleu metric**
ILP based Sentence Fusion	0.18	0.21	0.11
LDA topic Model	0.29	0.31	0.24
Sentence Clustering	0.36	0.35	0.31

From the above three tables, we have analysed that the News domain- Demonetisation data is best summarized by the Multi-document Summarization based on LDA Topic Model. For medical dataset and cricket dataset, Multi-document Summarization based on Sentence Clustering outperforms the other two algorithms. Sentence Clustering based algorithm gives better results because of its sentence clustering of single document summaries and extractive nature.

V. CONCLUSION

Simple single document extractive algorithms have given better results in different domains as compared to abstractive summarization algorithms.

Extractive summarizers are used to select the important set of sentences from the source document based on top scoring Sentence-ranking method. These methods use different feature extraction and content selection methods like upper case words, the frequency of words, similarity chains, logical closeness etc. for selecting summary sentences.

Abstractive Summarizers make new sentences by the union of multiple sentences. They use word graphs to select a set of words to produce a coherent sentence.

Based on the Comparison results, by performing Automatic Text Summarization to get a gist of the input text documents equivalent to human interpreted summary is not yet fulfilled, but by improving the existing algorithms, the value of evaluation metrics is increasing.

REFERENCES

[1] D.K. Gaikwad and C.N. Mahender "A Review Paper on Text Summarization", International Journal of Advanced Research in Computer and Communication Engineering Vol. 5, Issue 3, March 2016,154-160

[2] Radev, D. R., Hovy, E., and McKeown, K. (2002) "Introduction to the special issue on summarization." Computational Linguistics., 28(4):399-408

[3] Luhn, H. P. (1958) " The automatic creation of literature abstracts". IBM Journal of Research Development, 2(2):159-165

[4] Edmundson, H. P. (1969) " New methods in automatic extracting". Journal of the ACM, 16(2):264-285.

[5] R.Mihalcea, and P.Tarau, "TextRank: Bringing Order into Texts." In Proceedingsof Empirical Methods in Natural Language Processing (EMNLP). pp. 404-411. 2004.

[6] Z.Pei-ying, and L.Cun-he, "Automatic Text Summarization based on Sentences Clustering and Extraction," Proceeding of the 2nd IEEE International Conference on Computer Science and Information Technology. pp. 167-170. 2009

[7] 20IOy International Conference on Computer Application and System Modeling (ICCASM 2010) Automatic Text Summarization Based On Rhetorical Structure Theory Li Chengcheng 595-598

[8] D. Blei, A. Ng, and M. Jordan " Latent Dirichlet allocation". In Journal of Machine Learning Research, 3:993-1022, January2003.

[9] Barzilay, R. and Elhadad, M. (1997). "Using lexical chains for text summarization." in Proceedings ISTS'97. pg. 38-41

[10] Radev, D. R. and McKeown, K. (1998) "Generating natural language summaries from multiple on-line sources." Computational Linguistics, 24(3):469-500

[11] S. Banerjee, P.Mitra and K. Sugiyama " Multi-Document Abstractive Summarization Using ILP Based Multi-Sentence Compression"in Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)

[12] Lin, C.-Y. (2004). "Rouge: A package for automatic evaluation of summaries." In Proceedings of the ACL-04 Workshop, pages 74-81, Barcelona, Spain

[13] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu "BLEU: a Method for Automatic Evaluation of Machine Translation" in Computational Linguistics (ACL), Philadelphia, July 2002, pp. 311-318

[14] S. Brin and L. Page "The PageRank Citation Ranking:Bringing Order to the Web" in 1999

[15] Mc Keown, K. R. and Radev, D. R. (1995). "Generating summaries of multiple news articles." in Proceedings of SIGIR '95, pages 74-82, Seattle, Washington.

[16] Jagadeesh J, Prasad Pingali, Vasudeva Varma "Sentence Extraction Based Single Document Summarization" Workshop on Document Summarization, 19th and 20th March, 2005, IIIT Allahabad

[17] Kamal Sarkar, "Sentence Clustering-based Summarization of Multiple Text Documents", TECHNIA - International Journal of Computing Science and Communication Technologies, vol. 2, no. 1, Jul. 2009.

[18] F. Canan Pembe and Tunga Güngör, "Automated Query-biased and Structure-preserving Text Summarization on Web Documents," in Proceedings of the International Symposium on Innovations in Intelligent Systems and Applications, ?stanbul, June 2007.

[19] Reeve Lawrence H., Han Hyoil, Nagori Saya V., Yang Jonathan C., Schwimmer Tamara A., Brooks Ari D., "Concept Frequency Distribution in Biomedical Text Summarization", ACM 15th Conference on Information and Knowledge Management (CIKM), Arlington, VA, USA,2006.

[20] Khan Atif, Salim Naomie, "A review on abstractive summarization Methods", Journal of Theoretical and Applied Information Technology, 2014, Vol. 59

[21] Evans, D. K. (2005). "Similarity-based multilingual multi-document summarization." Technical Report CUCS-014-05, Columbia University.

[22] Edmundson, H. P. (1969). "New methods in automatic extracting." Journal of the ACM, 16(2):264-285.

[23] Martins, Camilla Brandel and Lucia Helena Machado Rino. "Revisiting UNLSumm: Improvement Through a Case Study." (2002).

[24] Conroy, J. M. and O'leary, D. P. (2001). "Text summarization via hidden markov models." In Proceedings of SIGIR '01, pages 406-407, New York, NY, USA

[25] Kupiec, J., Pedersen, J., and Chen, F. (1995). "A trainable document summarizer." In Proceedings SIGIR '95, pages 68-73, New York, NY, USA.

[26] Aone, C., Okurowski, M. E., Gorlinsky, J., and Larsen, B. (1999). "A trainable summarizer with knowledge acquired from robust nlp techniques".pages 71-80

[27] Lin, C.-Y. and Hovy, E. (1997). "Identifying topics by position." In Proceedings of the Fifth conference on Applied natural language processing, pages 283-290, San Francisco, CA, USA.

[28] Osborne, M. (2002). Using maximum entropy for sentence extraction. In Proceedings of the ACL'02 Workshop on Automatic Summarization, pages 1-8, Morristown, NJ, USA

[29] Svore, K., Vanderwende, L., and Burges, C. (2007). "Enhancing single-document summarization by combining RankNet and third-party sources." In Proceedings of the EMNLP-CoNLL, pages 448-457.

[30] Barzilay, R. and Elhadad, M. (1997). "Using lexical chains for text summarization." in Proceedings ISTS'97.

[31] Hovy, E. and Lin, C. Y. (1999). "Automated text summarization in summarist." In Mani, I. and Maybury, M. T., editors, Advances in Automatic Text Summarization, pages 81-94. MIT Press

[32] N. Aletras and M. Stevenson. "Evaluating topic coherence using distributional semantics." In Proc. Of the 10th Int. Conf. on Computational Semantics (IWCS'13), pages 13-22, 2013.

[33] Kamal Sarkar "Automatic Single Document Text Summarization Using Key Concepts in Documents" J Inf Process Syst, Vol.9, No.4, pp.602-620, December 2013

[34] I. Chen "Integer Linear Programming Models for Constrained Clustering" in International Conference on Discovery Science 2010: Discovery Science pp 159-173

[35] Günes Erkan and Dragomir R. Radev. 2004. "LexRank: graph-based lexical centrality as salience in text summarization". J. Artif. Int. Res. 22, 1 (December 2004), 457-479.

[36] Jinqiang Bian, Zengru Jiang, Qian Chen 2014 "Research On Multi-document Summarization Based On LDA Topic Model" Sixth International Conference on Intelligent Human-Machine Systems and Cybernetics 113-116

[37] Virendra Kumar Gupta Tanveer J. Siddiqui "Multi-Document Summarization Using Sentence Clustering" IEEE Proceedings of 4th International Conference on Intelligent Human Computer Interaction, Kharagpur, India, December 27-29, 2012

[38] The Porter Stemming Algorithm [Online] Available:http://tartarus.org/~martin/PorterStemmer/

[39] George A. Miller. "WordNet: A Lexical Database for English." Communications of the ACM, pages 39-41, November 1995

[40] Sherry and Dr. P. Bhatia " A Survey to Automatic Text Summarization Techniques" International Journal of Engineering Reasearch, October 2015 Pg. 1045- 1053

[41] Chin-Yew Lin and Eduard Hovy, "Identifying Topics by Position," In Proceedings of the Fifth conference on Applied natural language processing, San Francisco, pp. 283-290, 1997.

[42] S. P. Yong, A. I. Z. Abidin and Y. Y. Chen, "A Neural Based Text Summarization System," 6th International Conference of Data Mining, pp. 45-50, 2005.

[43] Ruqaiya Hasan, Coherence and Cohesive Harmony, In: Flood James (Ed.), Understanding Reading Comprehension: Cognition, Language and the Structure of Prose. Newark, Delaware: International Reading Association, pp. 181-219, 1984.

[44] William C. Mann and Sandra A. Thompson, Relational Propositions in Discourse, Defense Technical Information Center,

[45] Branimir Boguraev and Christopher Kennedy, "Saliencebased Content Characterization of Text Documents," In Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization, 1997.

[46] Li Chengcheng, "Automatic Text Summarization Based On Rhetorical Structure Theory," International Conference on Computer Application and System Modeling (ICCASM), vol. 13, pp. 595-598, October 2010.

[47] Xiaojun Wan, "An Exploration of Document Impact on Graph-Based Multi-Document Summarization," Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics,

[48] Tiedan Zhu and Xinxin Zhao, "An Improved Approach to Sentence Ordering For Multi-document Summarization," IACSIT Hong Kong Conferences, IACSIT Press, Singapore, vol. 25, pp. 29-33, 2012.

[49] Dhankhar, Yash, et al. "A Comprehensive Study of Text Summarization Algorithms.", IJSRCSEIT volume 4, issue 1, March - April 2018