ISSN: 2348 - 2117

International Journal of Engineering Technology and Computer Research (IJETCR)

Available Online at www.ijetcr.org

Volume 10; Issue 4; July-August; 2022; Page No. 01-10

A Comprehensive Survey of Scalable Multi-Document Summarization using Natural Language Processing

Aman Kaul[1], Vivek Rai[2]

[1]B.N College of Engineering and Technology, Lucknow, U.P. (India)

E-mail: amankaul1991@gmail.com

[2]Assistant Professor, B.N College of Engineering and Technology, Lucknow, U.P. (India)

Conflicts of interest: Nil

Corresponding author: Vivek Rai

Abstract

Natural Language Processing (NLP) approaches are the major objectives for users in this emergence of the Internet. Researchers have recently been interested in multi-document summarization (MDS) because of the difficulties it provides in producing well-summarized results. MDS is also an effective method for aggregating information since it creates a comprehensive and brief summary from a collection of topic-related papers. Document size limitations, limited memory in computer resources, and redundancy of identical words in numerous documents are all difficulties that MDS has to deal with. Natural Language Processing (NLP) methods are used in this paper to address these difficulties in MDS. Our study is the first of its type and provides a comprehensive overview of current NLP-based multi-document summarization methods via a suggested taxonomy. In addition, this paper offers a new approach for summarizing NLP design techniques and performing a systematic state-of-the-art review. The differences between various NLP approaches are highlighted, that are typically addressed in the existing literature. Furthermore, we discuss some future directions for this field's new and innovative advancement.

Keywords: Natural Language Processing; Multi-Document Summarization; Scalable; Data Extraction; Text Summarization.

I. Introduction

The ever-increasing volume of text documents finds evaluating and interpreting text a challenging task as technology advances in today's fast-paced world [1]. Getting the most important information out of a lot of documents is a time-consuming and work process from the reader's perspective. The vast amount of text data necessitates the use of text summarizing technologies to analyze the massive volumes more effectively [2]. Text summarization is an essential job in "Natural Language Processing (NLP)" which converts a text or a group of texts on similar issues into a comprehensive overview including crucial textual information. Summaries are often much smaller than the actual texts [3]. "Automatic Text Summarization" analysis has increasing volumes in the area of "Natural Language Processing" [4], and it can be useful for a variety of databases like making news digests, searching, and generating summaries [5].

Text summarization may be categorized into two stages based on the number of input documents: "Single Document Summarizing and Multi-Document Summarization". Multi-document

summarizing tries to generate a brief and useful summary from a group of subject-relevant texts, whereas "Single Document Summarization" proposes to construct a summary from only one document. Single document summary may not meet the criteria for complete summaries in terms of applicability since it does not make use of documents that are created continuously. It is more thorough and accurate to produce a summary from various documents written at various stages and from differing viewpoints to summarize data. Multi-document summarizing is more involved and harder to solve from a technological standpoint than a single-document summary. This is because there is more different and contradictory information across papers in the multi-document summary assignment. Documents generally have a larger volume and more intricate relationships between them. Articles would generally support, overlay, and interact with one another in such a huge number of documents [1]. Excessively prolonged input documents can also cause model deterioration [6]. Systems have a difficult task in retaining the most important elements of complicated input sequences while producing "coherent, non-redundant, non-factual error, and grammatically legible summaries". As a result, multi-document summarization needs more powerful models capable of evaluating corpora, recognizing, and combining consistent data. Moreover, due to the growing quantity of relevant datasets and language prediction models, the MDS job is becoming more computationally efficient.

Summarization on news today, published papers, emails, customer reviews, lecture responses, Wikipedia article era, medical documents, and software project tasks are only a few of the real-world applications for the multi-document summarization process. The multi-document summarizing technique has recently achieved a high level of interest in the market. The development of MDS is aided by large application needs and quickly expanding internet data. Moreover, the majority of previous techniques still use manually crafted features like "sentence position features, sentence length features, proper noun features, cue-features, biased word features, sentence-to-sentence cohesion, and sentence-to-centroid cohesion to develop summaries" [1].

NLP is a growing field of computational linguistics that evaluates, analyses, and replicates textual information using a variety of statistical methods. Learning methods like "pattern recognition, parts-of-speech (POS) tagging, and textual summarization" are used in NLP to ease human-computer interactions. NLP is essential for robots to comprehend and communicate with people. Researchers are concentrating on multi-document summarization (MDS) [7] in recent years because of the difficulties it poses in producing well-summarized findings. MDS is a summary of a collection of articles about a specific subject. Each year, the Text Analysis Conference (TAC) evaluates summarization models submitted by scholars. To be eligible for TAC assessment, a model must be able to reduce content redundancy and have a compression ratio of less than 10%. TAC assessment also requires the complexity of words retrieved by MDS in each document. There may be some redundancy in sentences from several articles when dealing with many entries on the same topic.

Both of these articles have high term frequency and sentence similarity, thus if the learning algorithm does not contain the necessary restrictions, both of them might end up in the summary. Reducing the number of a summary in MDS is also a difficult process; there may be differing perspectives in two articles on the same issue, both of which must be represented in the summary, which would significantly raise the file size. The size limits of papers, limited memory in computer resources, and redundancy of identical terms in numerous documents are all challenges that MDS has to deal with effectively. Using Natural Language Processing (NLP) methods, this paper overcomes these difficulties in MDS.

II. Multi-Document Summarization (MDS)

"Multi-Document Summarization (MDS) objectives to provide a brief and relevant summary *Sum* from a set of documents M. P $\{m_x | x \in$

$[1, B]\}$, signifies a cluster of topic-related documents, where B is the number of documents. Each document m_x is made up of A phrases $\{p_{x,y}|y \in [1, A]\}$. The y[th] phrase in the x[th] text is referred to as, $p_{x,y}$. The golden summary, often known as the reference summary, is a standard summary.

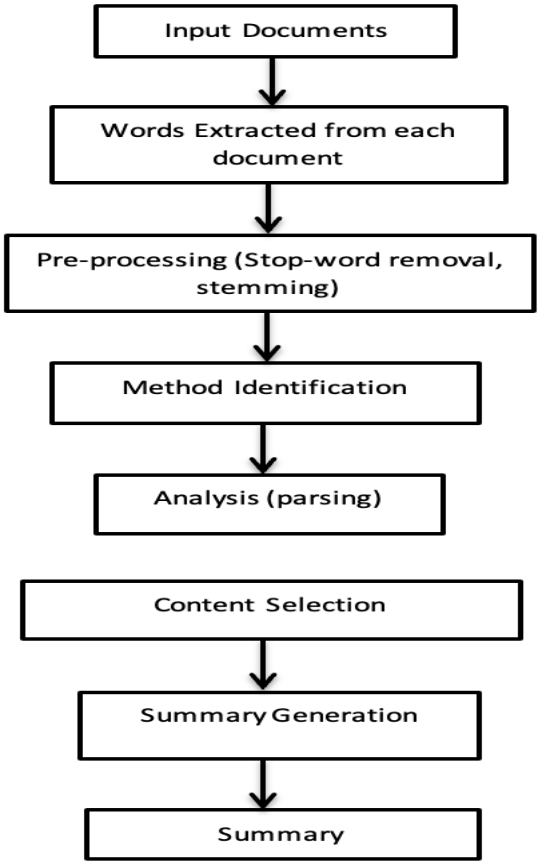

Figure 1: Basic Workflow of the Multi-Document Summarization (MDS)

This section describes and explains the processing structure in Figure 1 to give users a good survey of the processing of NLP-based multi-document summarizing job. The number of input documents is a major difference between "single document and multi-document summarization" at the input document level. The various types and lengths of input documents in the multi-document summarization process [1] are varied, and they may be classified into three classes:

Short Documents: The input data is huge, but the length of every document is quite small. Product reviews [8] are a common example of this sort of information. The goal of summarization on product reviews is to provide a brief and useful summary from a large number of customer evaluations of a product that reflects the general public's opinion.

Long Documents: All documents are significant, although there are a relatively limited number of them. For instance, creating a concise summary from a collection of news [9], or developing a "Wikipedia-style page" from a collection of online articles [10].

Hybrid Documents: One or more lengthy documents are combined with several smaller documents. Reader-aware multi-document summary, for example, incorporates news with many other readers' comments. Another example is

creating a technical overview from a huge scientific article with several brief relevant citations [11].

III. Related Works

Multi-document summarizing aims to condense a group of linked papers into a concise and useful summary. Existing multi-document summarizing approaches may be divided into two groups based on the forms of summary design:

3.1 Abstractive Multi-Document Summarization (AMDS)

This method is generating a few creative sentences which summarize the main ideas of an article or a piece of writing to create an eye-catching statement or a summary [12]. The authors behind this technique [13] have developed a way to enhance accuracy by feeding the encoder a collection of related sentences to check for the probability of generating inaccurate information. During their effort on identifying patterns to create summaries, they also generated emerging technologies for other NLP tasks. They got somewhat better results by using transformers for the grouping job, but it's a complex model with a high computational cost. Because the MDS job is their major focus, they haven't explored transformers for the clustering framework to ensure both efficiency and accuracy.

These researchers [14] proposed that concepts and situations are necessary for the semantic representation of records such as news articles. Occasions depict activities based on ideas, normally in the form of "who did what to whom when and where," addressing the major interest of readers of records such as news. A semantically Linked System of theories and situations is a good example of a Semantically Linked Network for communicating with the semantics of news articles documents. Summarization from numerous documents could be triggered by converting the data into a Semantically Linked Network of concepts and situations, decreasing the network to obtain a smaller and more logically correct network, and then altering the network to provide an effective textual summary.

Figure 2: Abstractive Multi-Document Summarization

Abstractive summarizing approaches aim to provide the most important information from input materials while dynamically creating short and coherent summaries. Techniques can create “new words and sentences from a corpus pool” using this set of approaches [15]. When compared to extractive summarizing algorithms, abstractive summarization's processing is more like that of human-written summaries. Abstractive summarization is difficult because it necessitates more advanced natural language interpretation and production methods, including paraphrase and sentence fusion [1].

3.2 Extractive Multi-Document Summarization (EMDS)

To create a new summary, extractive text summarization involves selecting expressions and sentences from the base text. Procedure entail ranking the importance of phrases to choose only those that are most relevant to the source's implication [12]. The authors [16] offer several architectures for parallelizing the MOABC computation in this study. Two irregular number generators, as well as the different calendars provided by the OpenMP framework, are being partitioned and considered for parallelization. Then, based on bee colony behavior, an asynchronous parallel structure was built. The experiments used the DUC datasets, revealing the powerful benefits of their technique from two aspects: computational execution and summary quality.

This technique [17] demonstrates how domain-specific characteristics may be quite useful in the process of summary creation. Various pre-processing stages are performed on raw texts in the proposed technique, and a useful feature vector is

generated based on domain-specific characteristics. This method [18] demonstrated the need to efficiently scavenging and sifting relevant information from the Internet. Along with obtaining necessary information, there is a requirement for effective content coverage with a wide range of information. Moreover, there is a lot to be done in the current summarizers' performance. The goal of the summarizing technique is to provide an accurate and continuous overview of the supplied document by verifying that the best significant parts of the elements are included while minimizing repetition from multiple input sources.

Figure 3: Extractive Multi-Document Summarization

To provide useful summaries, extractive summarization algorithms choose major samples from the original materials [19]. The two main components of these techniques are sentence ranking and sentence selection. The produced summaries are semantically comparable to the source documents thanks to extractive summarizing approaches. These approaches, however, confront several difficulties: “a) how to choose the most "meaningful" material; b) how to enhance the coherence and flexibility of generated summaries; and c) how to minimize duplicate information among selected phrases”.

IV. Natural Language Processing based Multi-Document Summarization Methods

The goal of “Natural Language Processing (NLP)” is for computers to interpret and analyze natural language processing and voice in a human-like way. “Part-of-speech (POS) tagging, named entity recognition, co-reference resolution, and machine translation” are examples of NLP tasks. Information extraction (IE) is a subset of natural language processing that seeks to extract specific information from text sources to fill in pre-defined information templates. In the specific area, NLP methods have been used in numerous research work for a range of applications [20].

4.1 Latent Semantic Analysis (LSA) Approach

LSA is an automated mathematical/statistical approach for extracting and describing the definitions of words in context in conversation sequences. The emerging theory is that the sum of all the word contexts in which a particular word appears and does not occur determines the similarity of meanings of words and groups of words. LSA has been applied to a wide range of circumstances [21].

The document-representation created in two phases is at the basis of the summarizing background analysis. The first stage is to develop a term by sentence matrix, in which each column represents a sentence's weighted term frequency vector in the set of documents being studied. A user query's terms are given more weight. The next step is to decompose matrix X using “Singular Value Decomposition (SVD)”:

$$X = I \sum J^{S} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

In terms of NLP, SVD is used to derive the text's latent semantic architecture, which is given by matrix A: that is, an analysis of the original content into r linearly-independent basis vectors that describe the document's key "issues." SVD may record interrelationships between concepts, allowing phrases and sentences to be grouped on a "semantic" rather than a "lexical" basis. Furthermore, as illustrated in [21], a significant and recurrent word combination pattern in a text will be collected and described by one of the singular vectors. The size of the related singular value denotes the pattern's significance in the document. Any sentences that include this word combination would be projected along this single vector, with the phrase that best depicts the pattern having the greatest index value. Assuming that each word combination pattern reflects a specific issue in a text, each singular vector may be thought of as describing that topic, with the size of its unique value indicating the topic's significance.

The technique chooses for the summary the phrases with the longest vectorial representation in the matrix, $\sum^2 . J^S$. The aim is to determine the phrases that have the most combined weight across all main aspects. The process of summarizing multi documents is one step more difficult than summarizing a single document. It introduces new issues that we must address. The 1st step is to make a term-by-sentence matrix once again. "All sentences from the cluster of documents" are included in this case's matrix. Following that, they rank the sentences. Each phrase is assigned a score, which is calculated in the same way as when summarizing a single text - vector length in the matrix, $\sum^2 . J^S$.

Let may now choose the most effective sentences for the summary. The "cosine similarity in the original term space" is used to classify the similarity. The user query must be close to the extracted sentence. To meet this requirement, question words are given greater weight in the input, $matrix^2$. Experiments with reduced dimensionality led to substantial improvements.

4.2 Probabilistic Latent Semantic Analysis (PLSA)

For word matching in retrieval applications, "Probabilistic Latent Semantic Analysis" [22] is a latent variable approach for co-occurrence data that has been proven to offer superior results than LSI [23]. It links each observation (x, a) to an unobserved class variable c $\in C = \{c_1, \dots, c_l\}$, where word a $\in$ A = $\{a_1, \dots, a_s\}$appears in document x $\in$ X = $\{x_1, \dots, x_t\}$. Every word in a text is represented as a sample from a probability distribution, with the mixture components being multinomial random variables which may be thought of as descriptions of latent issues. A document is reduced to a probability distribution over a fixed set of latent classes as a list of mixing proportions for the mixing factors.

PLSA may be defined based on a generative method as follows:

- Choose a document d that has a probability of P(x),
- Choose a latent class c with P(c|x) probability,
- P(a|c) is the probability of generating the term a.
- The resultant probability statement is for each observation pair (x, a):

"P (x, a) = P(x) P (a|x), where……(2)

P (a|x) = $\sum_{c \in C}$ P (a|c) P(c|x);"… .(3)

Considering an unknown topic 'c', a document 'x' and word 'a' are considered to be statistically independent. The mixing factors and proportions are chosen by maximizing the likelihood function, according to the maximum likelihood theory.

"S = $\sum_{x \in X} \sum_{a \in A}$ g(x, a) log $p(x, a)$ "…….(4)

Where, g(x, a) is the frequency value, i.e. the number of times "a" appeared in "x". The Expectation-Maximization (EM) algorithm is the typical approach for maximizing the likelihood function in the context of latent variables. EM is an iterative technique with the following stages: an expectation phase that evaluates the posterior probability for the latent classes "c" and a maximization phase that maintains the conditional probabilities of the factors given the posterior probabilities of the latent classes. By switching the prediction and maximization stages, one can achieve a convergence point that corresponds to a logarithmic probability local maximum. The mixture components, and also the mixing proportions over the modules for each training document, constitute the algorithm's output, i.e. the conditional probabilities P (a|c) and P(c|x). In [22 & 23] for further informations on the EM algorithm and its application to PLSA.

4.3 Latent Dirichlet Allocation (LDA)

LDA is a prominent topic modeling approach that represents text documents as combinations of latent topics, which major concepts are given in the text. A proposed approach is a probabilistic distribution approach used to a collection of text documents, in which each document is described as a collection of topics, which describe clusters of words that frequently appear simultaneously. The probability distribution across lexical words is modeled for

each topic. Each subject is represented as a vector of terms with probabilities ranging from 0 to 1. In LDA, a document is treated as a probability distribution across issues, with the topic mixture generated from the same conjugate Dirichlet basis for all texts [24].

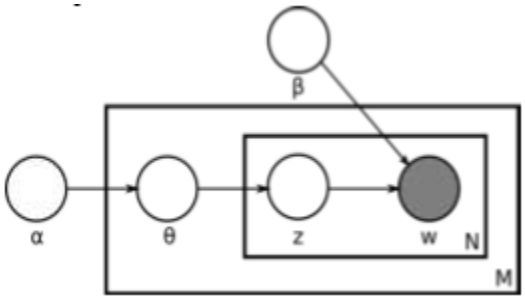

Figure 4: Graphical Representation of LDA Approaches

LDA uses Dirichlet features for the probabilities over a given number of topics to evaluate the topic-term distribution and the document topic distribution from an unlabeled collection of documents.

$$\iint \sum_{n=1}^{q} p(\sigma_n|\gamma) \sum_{y=1}^{M} p\,(\sigma_y|\varepsilon)$$
$$(\sum_{n=1}^{Yy} \sum_{y=1}^{Q} p(n_a|\sigma))\ \mathrm{p}(s_a|n,\mu)\ \mathrm{x}\ \sigma \mathrm{x}\ \mu \qquad (5)$$

Four phases are involved in topic modeling for text collecting by LDA. In the 1st stage a multinomial "σ_n" distribution, is chosen from a "Dirichlet Distribution" with variance "γ", for each issue "n". In the second stage, a "Multinomial σ_y Distribution is chosen from the Dirichlet Distribution with variance" for each document "x". In the third stage for each term w in documents, a subject t from "σ_y" is selected. And lastly, in the fourth stage, a term w from "σ_n" is picked to indicate the issue for the text document.

V. Results and Discussions.

Some papers used two data sets from recent summarizing challenges to evaluate our summarization system: "Multi-Document Summarization in DUC 2006 and DUC 2007" [22]. ROUGE metrics [25] are used in all of our evaluations. ROUGE measures are based on n-gram overlap and are recall-oriented. ROUGE-1 has been found to have a good correlation with generalization ability [26]. It also provides performance data for "ROUGE-2 and ROUGE-SU4" measures.

They used two baseline frameworks: Lead and an LSI-based system [25]. The lead phrases from the most current news report in the document cluster are used as the summary by the Lead system. The "rank-k singular value decomposition of the term-sentence matrix" is computed using the LSI baseline. The sentences in the latent semantic space are represented by the right-singular vectors that have been scaled by the singular values. Using the cosine similarity metric, then calculate the similar sentence-level features as the "PLSA-based system" and generate a summary via our greedy ranking and redundancy reduction method.

Participants in the DUC2006 multi-document summarization challenge are given 50 document clusters, each of which comprises 25 news items on the same topic. For each cluster, participants are required to provide summaries of no more than 250 words. A title and a narrative explaining a user's information requirement are supplied for each cluster. Typically, the narrative consists of a series of questions or a multi-sentence work description. The multi-document summary challenge in DUC-2007 is the same as it was in DUC-2006, with participants being required to develop 250-word multi-document summaries for 45 document clusters.

This section evaluates the proposed summarizing techniques and algorithms. The experimental analysis was carried out with two factors in mind: quality and quantity. The quality study compares the extracted summaries to human-derived summaries and current methods such lexRank and centrality, focusing on linguistic quality and readability. The quantitative tests assess the scalability and speed of the summary produced [7].

"ROUGE-1, ROUGE-2, and ROUGE-L values for each summary were calculated in our analyses. The following are ROUGE-N (1, 2) and ROUGE-L":

"ROUGE-N stands for n-gram co-occurences between candidate and reference summaries, with n being the number of words to match. They gather the ratio of the number of single words matched by the number of words in the reference summary in the 1-gram metrics. The ratio of two continuous words matching in both summaries is called 2-gram metrics". When summarizing several documents, the average of all the n-gram values is taken into account.

$$\text{ROUGE-N} = \frac{\sum_{M \in Ref.sum} \sum_{gram_x \in M} Count_{match}(gram_x)}{\sum_{M \in Ref.sum} \sum_{gram_x \in M} Count\,(gram_x)} \quad \ldots(6)$$

$$\text{Avg of ROUGE-N} = \frac{\pi A_x}{no.of.documents} \quad \ldots\ldots\ldots\ldots\ldots\ldots(7)$$

The most common evaluation measures are precision and recall. The "number of matching words in a candidate, reference summaries divided by the number of words in the candidate summary equals precision".

$$\text{Precision} = \frac{no.of.match}{no.of.words\ in\ cand.sum} \quad (8)$$

"Recall based models consider ratio of several matching n-units by several n-grams in the reference summary".

$$\text{Recall} = \frac{no.of.match}{no.of.words\ in\ ref.sum} \quad (9)$$

The F-measure is a statistic that is dependent on both precision and recall.

$$\text{F-Measure} = \frac{(1+\alpha^2)\, b*P}{r+\alpha^2*p} \quad \text{where } \alpha \leq 1. \quad (10)$$

Between candidate and reference summaries, Rouge-L considers the frequent patterns subset of word matching. One benefit of Rouge-L over n-gram ROUGE measures is that sentences with no continuous match words could still contribute to a summary score. Purpose of performing the evaluation, the n-gram doesn't have to be defined because it will choose continuous n-units by default. In terms of Precision, Recall, and F1 score derived by the appropriate formulae, NLP-based approaches were compared to existing algorithms. Table 1 shows the comparison of Evaluation metrics using Existing and proposed systems.

Table 1: Comparison of Evaluation Metrics using Existing with Proposed NLP Techniques

Models/Techniques	Precision (%)	Recall (%)	F-Measure (%)
LSA	85	86	82
PLSA	94	89	85
LDA	91	87	89

The proposed NLP is compared to existing algorithms in terms of "precision, recall, and F1 score". The graph is shown for the above-mentioned context, with the green bar representing F-measure, the red bar representing Recall, and the blue bar representing the Precision model.

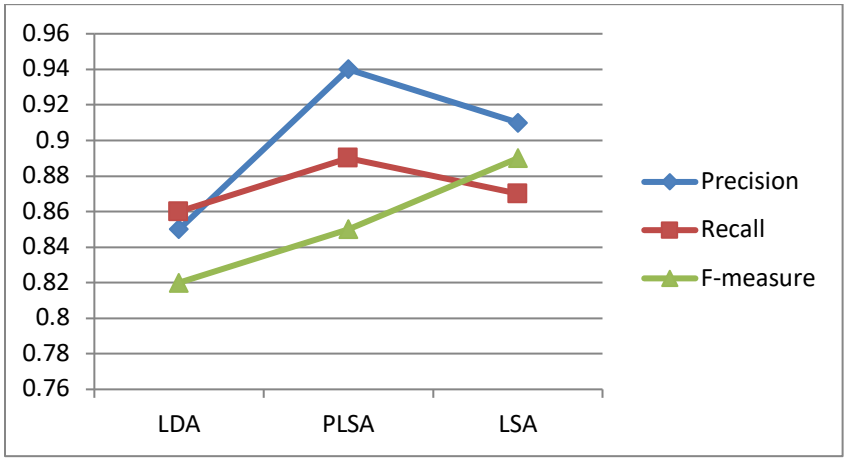

Figure 5: Graphical Representation of Comparison of NLP based Techniques

VI. Conclusion

Natural language processing is a field of languages, machine learning, and computer science whose objective is to allow humans and computers to interface using natural language. Even now, there is no standard methodology for text summarization. This study covers the benefits and drawbacks of different existing techniques in the hopes that future researchers may be able to design more efficient or hybrid ways based on the outcomes of the techniques described. Using a basic unsupervised method, we were able to get these extremely competitive results. When compared to a system that uses latent semantic indexing, the PLSA approach incorporates the sparse information in a sentence better than a comparable LSA or LDA approach. Modern techniques or a combination of two or more techniques could be developed in the future with the help of natural language processing and language methods, which could be utilized to produce better multi-document summaries.

References

1. Ma, C., Zhang, W. E., Guo, M., Wang, H., & Sheng, Q. Z. (2020). Multi-document Summarization via Deep Learning Techniques: A Survey. arXiv preprint arXiv:2011.04843.
2. Hu, Y. H., Chen, Y. L., & Chou, H. L. (2017). Opinion mining from online hotel reviews–a text summarization approach. Information Processing & Management, 53(2), 436-449.
3. Peyrard, M. (2018). A simple theoretical model of importance for summarization. arXiv preprint arXiv:1801.08991.
4. Moratanch, N., & Chitrakala, S. (2016, March). A survey on abstractive text summarization. In 2016 International Conference on Circuit, power and computing technologies (ICCPCT) (pp. 1-7). IEEE.
5. Paulus, R., Xiong, C., & Socher, R. (2017). A deep reinforced model for abstractive summarization. arXiv preprint arXiv:1705.04304.
6. Jin, H., Wang, T., & Wan, X. (2020, July). Multi-granularity interaction network for extractive and abstractive multi-document summarization. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (pp. 6244-6254).
7. Prabhala, B. (2014). Scalable Multi-Document Summarization Using Natural Language Processing. Rochester Institute of Technology.
8. Angelidis, S., & Lapata, M. (2018). Summarizing opinions: Aspect extraction meets sentiment prediction and they are both weakly supervised. arXiv preprint arXiv:1808.08858.
9. Fabbri, A. R., Li, I., She, T., Li, S., & Radev, D. R. (2019). Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. arXiv preprint arXiv:1906.01749.
10. Liu, P. J., Saleh, M., Pot, E., Goodrich, B., Sepassi, R., Kaiser, L., & Shazeer, N. (2018). Generating Wikipedia by summarizing long sequences. arXiv preprint arXiv:1801.10198.
11. Yasunaga, M., Kasai, J., Zhang, R., Fabbri, A. R., Li, I., Friedman, D., & Radev, D. R. (2019, July). Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 33, No. 01, pp. 7386-7393).
12. Yash Asawa, Vignesh Balaji, Ishan Isaac Dey. (2020). Modern Multi-Document Text Summarization Techniques. International Journal of Recent Technology and Engineering (IJRTE). 9(1), 654-670.
13. Qiang, J. P., Chen, P., Ding, W., Xie, F., & Wu, X. (2016). Multi-document summarization using closed patterns. Knowledge-Based Systems, 99, 28-38.
14. Fuad, T. A., Nayeem, M. T., Mahmud, A., & Chali, Y. (2019). Neural sentence fusion for diversity driven abstractive multi-document summarization. Computer Speech & Language, 58, 216-230.
15. Paulus, R., Xiong, C., & Socher, R. (2017). A deep reinforced model for abstractive summarization. arXiv preprint arXiv:1705. 04304.

16. Li, W., & Zhuge, H. (2019). Abstractive multi-document summarization based on semantic link network. IEEE Transactions on Knowledge and Data Engineering.
17. Sanchez-Gomez, J. M., Vega-Rodríguez, M. A., & Perez, C. J. (2019). Parallelizing a multi-objective optimization approach for extractive multi-document text summarization. Journal of Parallel and Distributed Computing, 134, 166-179.
18. Mutlu, B., Sezer, E. A., & Akcayol, M. A. (2019). Multi-document extractive text summarization: A comparative assessment on features. Knowledge-Based Systems, 183, 104848.
19. Nallapati, R., Zhai, F., & Zhou, B. (2017, February). Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In Thirty-First AAAI Conference on Artificial Intelligence.
20. Zhang, J., & El-Gohary, N. (2012). Extraction of construction regulatory requirements from textual documents using natural language processing techniques. In Computing in Civil Engineering (2012) (pp. 453-460).
21. Steinberger, J., & Křišťan, M. (2007). Lsa-based multi-document summarization. In Proceedings of 8th International PhD Workshop on Systems and Control (Vol. 7).
22. Hennig, L. (2009, September). Topic-based multi-document summarization with probabilistic latent semantic analysis. In Proceedings of the International Conference RANLP-2009 (pp. 144-149).
23. Hofmann, T. (1999, August). Probabilistic latent semantic indexing. In Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval (pp. 50-57).
24. Gowri, K., & Chezian, R. M. AN IMPROVED TEXT SUMMARIZATION USING FEATURE SELECTION AND OPTIMIZED NAIVE BAYES CLASSIFICATION COMPARED WITH LATENT DIRICHLET ALLOCATION.
25. Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. the Journal of machine Learning research, 3, 993-1022.
26. Lin, C. Y., & Hovy, E. (2003). Automatic evaluation of summaries using n-gram co-occurrence statistics. In Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (pp. 150-157).