IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Extractive Multi-Document Summarization: A Review of Progress in the Last Decade

Zakia Jalil[1], Jamal Abdul Nasir[1], and Muhammad Nasir[1]
[1]International Islamic University, H-10, Islamabad, Pakistan

Corresponding author: Zakia Jalil (e-mail: zakia.jalil@iiu.edu.pk)

ABSTRACT With the tremendous growth in the number of electronic documents, it is becoming challenging to manage the volume of information. Much research has focused on automatically summarizing the information available in the documents. Multi-Document Summarization (MDS) is one approach that aims to extract the information from the available documents in such a concise way that none of the important points are missed from the summary while avoiding the redundancy of information at the same time. This study presents an extensive survey of extractive MDS over the last decade to show the progress of research in this field. We present different techniques of extractive MDS and compare their strengths and weaknesses. Research work is presented by category and evaluated to help the reader understand the work in this field and to guide them in defining their own research directions. Benchmark datasets and standard evaluation techniques are also presented. This study concludes that most of the extractive MDS techniques are successful in developing salient and information-rich summaries of the documents provided.

INDEX TERMS abstractive summarization, clustering, extractive summarization, graph-based, machine learning, multi-document summarization, natural language processing, ontology, term-based

I. INTRODUCTION

Since the emergence of computers, the reliance of individuals and companies on computers has increased at a remarkable pace. With the invention of the internet, this reliance became even more evident. The amount of data and information stored on disks started increasing. Today, the extraction of information from such a huge amount of data is a tedious task, generally associated with information overload [1]–[4]. In order to access information in minimum time, it is necessary to represent the information in a more compact format. Automatic Text Summarization (ATS) is one of the solutions that address this need; ATS is deeply rooted in the history of text summarization for over five decades [5], [6]. Text summarization is the process of extracting information in such a way that the valuable information is not missed out in the generated summary, yet avoiding the redundancy of the original format [1], [2].

As text summarization eliminates redundant data from digital documents, it has been used to facilitate computer use by people with different medical disabilities. One such case is text summarization by Barbu et al. [7] for people with Autism Spectrum Disorder (ASD). Similarly, researchers from languages other than English have also benefited from the techniques used for multi-document summarization by using it in their respective languages. One such example is Oufaida et al. [8], who used Minimum Redundancy and Maximum Coverage algorithm (mRMC) for Arabic text.

Text summarization can be divided into two broad categories [9], namely single-document summarization and multi-document summarization. Single-document summarization is the process of extracting the most significant information from a document in a concise format for ease of readability [9], [10]. Multi-document summarization handles cases where the information is spread over multiple sources and documents. For instance, the same contents may be covered from multiple sources, so at times, a number of documents may be available to gain an insight into the same event [5]. In this regard, a multi-document summary becomes a representation of the information contained in a cluster of documents which helps users understand the gist of those documents [9], [10]. A multi-document summary represents the information contained in the cluster of documents and helps users understand those documents [11].

The task of multi-document summarization is much more complex than single-document summarization, even when the available single document is very large-sized. This difficulty is attributed to the inevitable diversity of themes within a large set of documents.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112496, IEEE Access

A summary can be Abstractive or Extractive, depending on the method of summarization. Generally, an abstractive summary consists of concepts and ideas abstracted from the source document(s) and then represented in preferably different words. This method involves a thorough understanding of the meaning of the content. Two broad areas of Natural Language Processing (NLP) [12] that handle abstractive summary are semantic representation and natural language generation [4]. These involve various approaches, such as multimodal semantic models, information item-based methods, and semantic graph-based methods [13].

An extractive summary is described as units of text extracted from the source document(s) and combined as a summary verbatim [14]. In this method, the important sentences of the documents under consideration are ranked and combined to form the summary [15]. Extractive summarization techniques can be divided into various categories, such as: query-based or generic and supervised or unsupervised methods. Generic summarization is based on preparing a summary of the main topic of the documents, whereas query-based summarization involves generating a summary related to the subject of the query asked by the user [9], [10], [12], [16]–[21].

In order to gain a broader picture of research in this field, we have performed a systematic survey of the literature on extractive techniques of MDS. The survey may serve as a starting point for prospected researchers to identify gaps in current research. The paper is organized as follows: Section II presents a detailed category-based literature review; Section III gives details of different datasets used in papers included in this survey; Section IV sheds light on various evaluation techniques, section V is dedicated for discussion, and Section VI concludes the paper.

II. LITERATURE RIVIEW

Many multi-document summarization systems are available in the literature. Methodology-wise, extractive summarization is divided into Cluster-based techniques and Graph-based techniques [6]. The cluster-based method was first presented by Radev et al. [22] basic idea of which was to group similar sentences from the document(s) into clusters, and then choose the most salient sentences from each cluster to compile a summary of the document(s) [22], [23]. Radev et al. used tf-idf (Term Frequency-Inverse Document Frequency) based features in k-means clustering algorithm to group similar and salient sentences together. Tf-idf scores the importance of words (or “terms”) in a document based on how frequently they appear in multiple documents. If a term is frequent in all the documents, tf-idf ranks it low, assuming that it is not a salient term of a specific document; instead, it is a common term. Tf-idf helps to filter out closed-class words that are used frequently in a language but are not representative of the meaning of the document. The summarization produced using cluster-based methods brings in diverse information from documents and, at the same time, reduces the redundancy of data. Famous cluster-based summarization techniques are presented in [24]–[27]. The graph-based approaches [28], [29] uses the idea of the well-known PageRank algorithm [30], which was traditionally used in Web-link analysis and social networks. They build the sentence graph, and then their neighbors vote to select a sentence for the next vertex. The fundamental graph-based theory is maintained by the links between sentences existing based on some similarity values calculated by some techniques (like cosine similarity measure [31] between sentences. Sentence similarity is calculated in terms of other sentences in the documents. Sentences with high similarity values are considered best for summary sentence selection. The graph-based method is used in multi-document summarization to identify significant sentences among multiple documents [32]–[35]. However, if we consider the graph-based approach’s primary idea, sentences are connected based on a similarity value instead of relationship type [6].

While taking the latent semantics of the contents of the documents in a view, several methods are devised based on latent semantic analysis [36] and non-negative matrix factorization [37], [38]. Similarly, keeping in view the lexical semantics [16], ontology-based approaches [39], [40] have been used to produce summaries. Ensemble-based technique was also tested [41] for multi-document summarization, while Rhetoric based summarization has also been considered for the same purpose [42], [43].

Based on the literature studied, there are several widely used extractive summarization methods. Some of the categories are stated as follows:

A. ONTOLOGY-BASED METHODS

Ontologies are formal representations of the most unusual concepts related to a specified knowledge domain and different corresponding relationships. They are used in numerous research fields, including user-generated content analysis, e-learning framework development, video analysis, and image analysis. Recently, the use of ontologies is increased by the research community [17], [39], [40] due to its promising results in various fields, specifically in document summarization. It helps identify important sentences from the documents to generate a summary by incorporating ontological knowledge. Ontologies are used to show the document set’s critical concepts and their correlation with the user query by avoiding ambiguities.

An ontology-based approach was proposed by Baralis et al. [39], called YAGO summarizer, which used Wikipedia for mapping of words to non-ambiguous ontological concepts called entities. YAGO summarizer selects sentences from a document as per previously assigned entities.

This technique’s achievement is the use of ontology of a domain, which consequently eliminates the problems of synonymy and polysemy in multi-document summarization. The limitation of ontology-based approaches is that the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112496, IEEE Access

ontologies are domain-specific. Similarly, much of the efforts are needed to develop an ontology of some domain [16].

B. TERM- BASED METHODS

The term-based methods usually implement the bag-of-words (BOW) model to calculate the weight of a term using the tf-isf (Term Frequency-Inverse Sentence Frequency) weighting model and some variants of this scheme.

Oliveira et al. [4] presented a comparative analysis of eighteen shallow sentence salience-scoring techniques to compute a sentence's significance in extractive single and multi-document summarization. Numerous experiments were performed to evaluate the performance of these sentence-scoring techniques individually and applying different combination strategies over the news domain datasets of CNN Corpus and DUC 2001-2004. The sentence scoring techniques used various combinations of features like Word frequency, Word co-occurrence, Upper case, TextRank, tf-isf, Sentence resemblance to the title, Position of the sentence, Length of the sentence, Centrality of the sentence, Proper noun, Open relations, Numerical data, Noun and verbal phrases, Named entities, Lexical similarity, Cue-phrases, Aggregate similarity, and Bushy path. These scoring techniques were used as input features for different machine-learning algorithms provided by Weka toolkit, like *AdaBoostM1, J48, K-nearest Neighbours referred as IBK, Multilayer Perceptron, Multinomial Logistic Regression (Logistic), Naive Bayes, Random Forest, Random Tree, Radial Basis Function Network (RBFNetwork), and Support Vector Machines using Sequential Minimal Optimization (SMO)*. The state-of-the-art techniques for single-document summarization selected were Autosummarizer, Classifier4J, and HP-UFPE Functional summarization, along with the best performing participants of DUC 2001, 2012, while for multi-document summarization, the state-of-the-art systems were ICSISUMM, Greedy-KL, LLRSum, ProbSum, Sume, as well as the best performing participants from DUC 2001-2004 competition. It was observed that in combination with state-of-the-art, these techniques produce better results, but the standalone performance of these techniques is a bit compromised.

Another technique, named Maximum Coverage and Less Redundancy (MCLR) [9], represents multi-document summarization as a quadratic boolean programming problem to solve the optimization problem. In this method, a weighted combination of the content coverage and redundancy objectives are used to map the objective function [21].

A bottom-up approach was presented by Bollegala et al. [44] for arranging the sentences. To find the association between two sentences and obtain their order, they devised criteria based on chronology, topic-closeness, precedence, and succession [16].

The ordering of information in the generated summary plays a significant role. This need is iterated in [45], where vital sentences from the given set of documents are extracted first. This extraction is based on five characteristics, namely, chronology, probabilistic, topic-closeness, precedence, and succession. The meaningful extracted sentences are then arranged to add to the beauty of the summary. This ordering is done by using human-annotated summaries in the system. Once the system learns the best combinations, the model is tested on the automatically generated summaries. The proposed sentence ordering algorithm operates on pair-wise comparisons of sentences to determine the overall ordering. This is done using a greedy search algorithm that avoids the combinatorial time complexity, which is typically associated with total ordering tasks. This helps in quick sentence-ordering in more extended summaries; therefore, this approach is feasible for real-world text summarization systems.

Nasir et al. [46] used a measure of semantic relatedness, named Omiotis, to construct a flattening matrix and a kernel for semantically adjusting the BOW illustration. Omiotis is made from the thesaurus and WordNet (word dictionary), which handles the problem of synonymy and polysemy. Omiotis works on sense-related measure SR. It uses the BOW approach by embedding Omiotis into a semantic kernel. The recommended measure includes the tf-idf for producing a semantic kernel by combining the semantic and statistical information related to the text. It handles the word synonymy and polysemy problems. The Latent Semantic Analysis, discussed in detail in sub-section 2, helps handle the problem of synonym, but polysemy is yet to be resolved.

Bayesian topic modeling also has been used for document summarization [47]. Sentences in the document contain many embedded topics that are not focused on most of the summarization techniques. More importantly, it emphasizes the hidden embedded topics present in sentences to generate an appropriate and precise summary. Considering this method, it can be concluded that topic modeling helps in understanding the context by selecting the appropriate sentence, which would help generate an effective summary by makes use of both text document and the word sentence relationship.

The comprehensive comparison of all the term-based methods is presented in Table I.

The further categories of the term-based method are as follows:

1) CLUSTERING-BASED METHODS

Based on a set of features, clustering-based methods compute the similarity between sentences, also known as the salience of sentences, to rank them. MEAD [22] is an example of a clustering-based method that is used for sentence extraction. This task is done with three parameters, namely, the value of centroid (the average cosine similarity between sentences and the rest of the sentences in the documents), positional value(documents contain N sentences, leading sentences is given 1 as a score and for each sentence

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112496, IEEE Access

TABLE I
STRENGTHS AND WEAKNESSES OF TERM-BASED METHODS

Sr.#	Research Study	Working	Results & Evaluation
1	Oliveira et al., 2016	Eighteen shallow sentence scoring techniques are compared on different methods in the news domain. It is applied on SDS as well as MDS using an extractive method of summarization.	Individual results of these sentence scoring techniques are not promising. When combined with state-of-the-art methods, these techniques show comparable results.
2	Alguliev, Aliguliyev, & Hajirahimova, 2012	MCLR technique is presented. Sentences are scored as per features, and then prominent sentences are compared with each other. Unique ones are included in the summary. For optimization, a modified Differential Evolution algorithm is used.	Individual results of these sentence scoring techniques are not promising. When combined with state-of-the-art methods, these techniques show comparable results.
3	Bollegala et al., 2010	A sentence association and ordering technique is presented based on the criteria of chronology, topic closeness, precedence, and succession.	The algorithm is tested on a dataset of Japanese newspapers. However, it needs to be tested on standard benchmark datasets.
4	Bollegala et al., 2012	Probabilistic criterion is added to the work presented in Bollegala et al., 2010	It was only tested on the Japanese News dataset. Furthermore, testing is required on benchmark datasets.
5	Nasir et al., 2011	Omiotis measure of sense relatedness is used alongside the BOW approach to handle synonymy, polysemy, and word semantic relatedness problems.	In pre-processing of the data, stemming was missed. The use of stemming would have resulted in the omission of the similar root words used in separate ways due to grammatical reasons.
6	D. Wang et al., 2009	Bayesian method of topic modeling is presented for understanding the context of sentences. tf-isf is used for sentence ranking.	LDA uses exceptional maximization that increases the complexity and slows down optimization. Pre-processing of the data using deep natural language analysis is missing.

the score decreases with the ratio of 1/N), and finally the first-sentence overlap (the cosine similarity of a sentence with the first sentence in the same document). The three parameters are linearly combined and assigned equal weights. Figure 1 describes the clustering-based methods in detail.

Density-Peak Clustering Sentence (DPCS), proposed by Zhang et al. [48], calculates the sentence representativeness score and diversity score. It first calculates the sentence similarity matrix by dividing documents into sentences and then removing the stop words. After that, the sentences are represented as a bag of words, and a cosine comparison is calculated. The boolean system is used to assign weights to the sentences, and the representativeness score is calculated. Representativeness score describes the sentence that is important in the document. After that diversity of the sentences is calculated. Diversity score condenses the redundancy, which was the task of the post-processing unit. It is calculated by computing the minimum distance between some sentence i and the other sentence having the highest diversity. Length score helps to make the sentence length shorter. Real length is the number of words in a sentence, whereas effective length refers to the number of unique nonstop words in a sentence, i.e., the sum of unique words. The squatter sentences with better representativeness are extra ideal over those with long length. Experiments were done on datasets of DUC 04. It is therefore confirmed that the density peaks gathering method can effectively handle multi-document summarization. However, this work is at an initial stage and is open for further research inputs.

Wang et al. [49] presented Density-Peak based clustering technique for generic extractive multi-document

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112496, IEEE Access

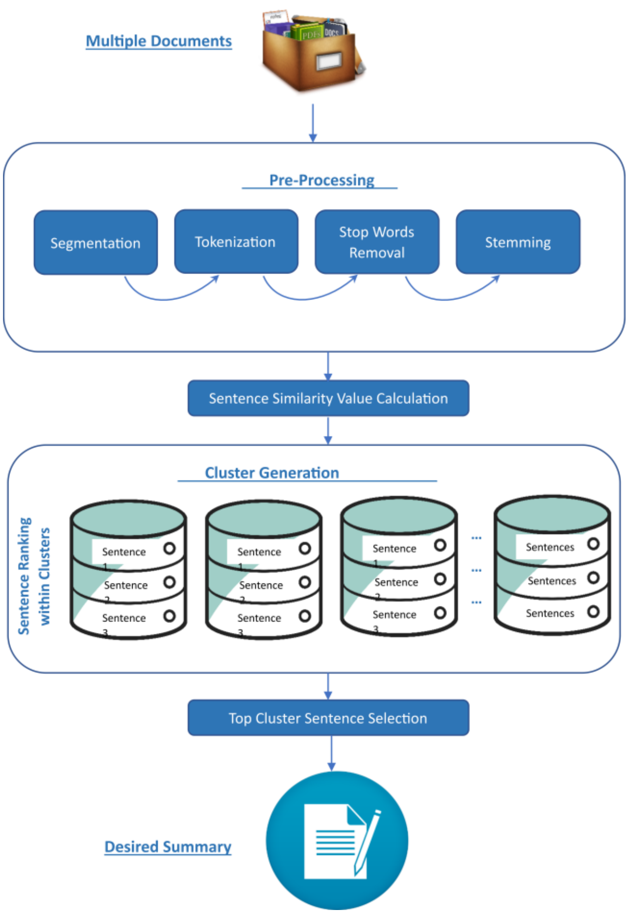

FIGURE 1. Clustering-based multi-document summarization

summarization. The benefit of Density-Peak's technique is that it does not demand to set the number of desired clusters in advance and is handled at run time. In clusters, sentences are ranked using Integrated Score Framework, and salient sentences are selected to be part of the summary using dynamic programming. This technique performed well in the ROUGE SU4 matric for summary evaluation, while in ROUGE 1 and ROUGE 2, its performance was not better than other techniques. Similarly, this technique did not handle the problems of synonymy and multi-vocal words in this work.

Nagwani et al. [50] worked on Big Data Analysis and presented the summarization of large data available in it. This is accomplished using topic modelling and semantic similarity clustering. This work is done in four stages. First, text clustering is used on the documents to create clusters via K-means so that similar documents contribute to the summarization task. In the second phase, Latent Dirichlet Allocation (LDA) creates the topics from given sentences. In the third stage, frequent word generation is done by sending the topic words (terms) produced from the LDA to the summarizer, then mixed up and transmitted to the mappers. Topic-terms frequency is computed, and frequently occurring terms are produced. After that, semantically similar terms for the frequent terms are produced with the help of WordNet. In the last stage for each document, sentence filtering is performed based on semantically identical words and frequent words. Sentences are picked from every document for frequently occurring words and their semantically similar words to constitute the summary. Duplicate sentences are removed, and a summary is generated. The MapReduce implementation collects all values linked with the same key and combines them in the reducer. The result of the algorithm is obtained in the distributed file system, having a file per reducer. In the end, the sentences containing the frequent terms are selected that will produce the summary of the given text. This is quite a detailed solution but a very costly one in the context of MDS. This work is done for big data analysis and inherits the drawbacks of the K-means algorithm, employing extensive external sources.

Christensen et al. [51] explored hierarchical summarization where the sentences at the top level provide an overview of the documents so that more information can be obtained by directing them into sentences. This sorts parent-child consistency and gives important information as per the attention of the user so that the user having particular interest can dig down into the information of its interest. In this way, the root sentence gives a general overview of the summary. By selecting an additional sentence of the summary, it gives more detail about the occasion. If the third sentence is selected, it will further provide information to go into depth and gain more details. In this way, every non-leaf node provides further details of the leaf nodes, i.e., a child gives more detail about the parent. Sentences were summarized by a technique named SUMMA summarizer implemented by Christensen et al. [51]. SUMMA uses articles and then combines the sentences forming the cluster with objective function concerning time, which further works on salience and coherence information. In hierarchical summarization, input is a set of related documents. There is a budget for each summary. The output is hierarchical summary and set of summaries. Child summary gives more details to the information, i.e., events or any other background. Each summary should have coherence which comprises of parent-child consistency and intra-cluster coherence. Initially, the quantity of information is less, and the user directs it as a topic of concern. Process of the summary generation is shortened into two parts. The first one is to create clusters and the second is summarization inside the clusters. Hierarchical clustering results in the clustering based on chronology. Then summarization of the gathered documents cluster is performed chronologically. Clustering algorithm is used recursively to choose the number of clusters which are time stamped prior to the gathering. Sentences are then parsed with Stanford parser. Documents are drawn to the topic by a sentence to topic value called salience. It adds the saliencies of individual sentences. Training of dataset was done with linear regression classifier, which is also used for identification of redundant sentences.

The features include shared noun counts, sentence length, tf-idf cosine similarity, and timestamp difference. In this regard, two types of coherence are required here, one is the parent-child coherence, and the other is coherence within each cluster. Therefore, an approximate discourse graph (ADG) is used for calculating coherence. In parent-child coherence, the user will move from the parent sentence to the child sentence, so there must be a proper link among parent-child sentences, and the sum of positive weight from parent sentence to a child will be displayed in ADG. In intra-cluster coherence, the summary is deemed acceptable if it has positive evidence in ADG. For calculating the quality of summary produced, a function is used that combines consistency, salience, and redundancy. Therefore, the number of sentences in summary must match the non-leaf cluster. The concern is that it deals with redundancy and budget as hard constraints while considering coherence and salience as soft constraints. It is based on timestamps and is location focused.

Clusters with random shapes can simply be noticed employing local density methods. It adapts the K-medoids technique. In an algorithm by Rodriguez et al. [52], cluster centers are enclosed by low local compactness neighbors. They have comparatively large space from any points with a higher local density. For each specific point, two modules are calculated: local density, and the other is the points with higher density. In local density, those points that are not close to Dc are cut down. Dc is the value that shows that point that is not closer to the distance between the data points Dij will be removed.

The other parameter is computed by discovering the least distance of point i from all the other points with higher density. If this parameter has a large irregular value, then it is measured as the cluster center. The algorithm has no noise cut-off. First, the border region of the cluster is defined. These will be the points that are assigned to the cluster. These points have a distance Dc from points that belong to other clusters. For each cluster, the point with the highest density is selected from the border region. The points above this value are considered part of the cluster core, and the other points are considered noise. This algorithm gets the position and shape of the clusters, which have even different densities. From many points, reduced samples are gained, and cluster assignment is performed in it. The wrong classified points' fraction remains below 1 percent, even for small samples containing 1000 points. In some cases, the datasets with a small number of points might be affected by significant statistical errors.

The work presented in [53] focused on Argumentative Zoning used for extractive summarization in the scientific domain. A trained classifier is used along with a feature-based clustering technique. The classifier's job is to create a preliminary candidate set of sentences to be included in the summary. The sentence cluster is used for identifying groups of connected (similar) sentences in that set created by the classifier. These groups are then used to generate the final summary. Clustering improves the quality of summary by removing redundancy from the candidate set. Sentences from training articles are pre-processed, labelled, stop words are removed from sentences, and lemmatization is done. After that, sentences are represented as a feature vector for the training of the classifier. The compression ratio and the number of clusters are threshold values set by the user. After classification, cluster generation is used for summary generation. The classification uses set A to be a set of sentences in the abstract of papers and set M to be a set of sentences in the paper's main body. Using sentences in sets A and M, the classifier is trained to generate sentences in set C, which is a set of sentences in summary. The sentences in set A are positively labeled, while set M's sentences can be positive or negative. Here the non-traditional classifier-based method is used for training. Artificially generated data can be used to train the classifier. The features are verb features, tf-idf, citations and reference occurrence, argumentative zones, and locative features. It means that previous work is present at the start of the information and future work is present at the end. The summary is supposed to provide comprehensive information of related work of the topic and its methodology. After sentence classification, K-means clustering is used to remove redundancy and identify similar sentences, and the desired summary is generated using cluster centroid. Another sentence clustering method is to group the sentences having the same argumentative zones label for easy identification of clusters. As per user requirements, the system can produce full-document and customized-document summaries. Hence, the conclusion is that the argumentative zone helps in producing effective summaries of the scientific domain. The problem here is that positive and negative labelling of sentences is complex, and clustering and classification make it a little costly solution.

In clustering-based methods, the technique by Christensen et al. [51] produces the best Rouge-1 values with Recall as 0.67 on DUC 2004

Table II sums up the pros and cons of the clustering-based method for a glance.

2) LATENT SEMANTIC ANALYSIS (LSA) METHODS

Gong and Liu [36] presented a technique consuming Latent Semantic Analysis (LSA) for ranking high-scoring sentences in the document collection for a summary generation. As shown in Figure 2, it creates a matrix of terms and sentences, where the columns show the weighted term-frequency vector of a sentence in the documents set. The latent semantic structure is then derived by using Singular Value Decomposition (SVD), which is a mathematical method to show the relationship among terms and sentences, on the input matrix. Different topics are identified in the document set, and those sentences having higher combined weights in all the topics are selected in summary.

IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

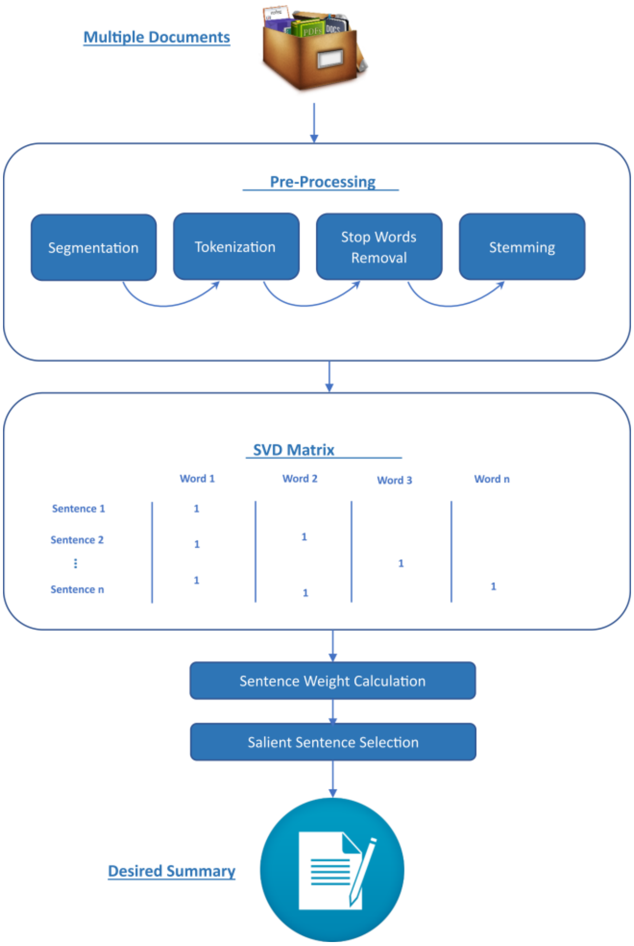

FIGURE 2. **Latent Semantic Analysis (LSA) based multi-document summarization.**

Another attempt to improve sentence similarity techniques was made by Ferreira et al. [54], who undertook sentence similarity and word order in their work. According to the authors, the following factors were not considered by the research community thus far: The Problem of Meaning: There are ways of writing sentences referring to the identical meaning written differently. Like the sentences “John is a handsome boy” and “John is a good-looking lad,” they have similar meanings if used in the same context. The Problem of Word Order: The order of the appearance of words in a text affects its meaning, like the same combination of words but a different order of words in the sentences “A killed B” and “B killed A” bring different meanings. Ferreira et al. [54] represented sentence in three layers, namely, (i) lexical layer that includes lexical analysis, stemming, and stop words removal (ii) syntactic layer to performs syntactic analysis, and (iii) semantic layer that mainly defines the semantic role annotations. This paper also presents a new similarity measure between sentences. The text semantics are obtained using semantic role annotation (SRA), which previously were obtained using WordNet. The three-layer sentence representation handles the problems of meaning and word order.

The event-based technique was used by Marujo et al. [55]. In this work, event information and word embeddings are used in multi-document summarization. KPCentrality method that is already used in a single document summarization was extended for multi-document summarization. It was used in single layer as well as waterfall approaches. The single-layer approach generates a summary by adding the summaries of every input document at the end. On the other hand, the waterfall approach joins the summaries of every input document based on a timestamp of documents in a cascade style. Event information is used in the filtering stage and the improvement of sentence representation.

Maximum Marginal Relevance (MMR) [56] combines query-relevance and innovation criteria to remove redundancy. In the end, the dissimilarity is computed among the documents in the ranked list. MMR considers the relevant novelty, which can be calculated independently for the ranked documents. The text will only have high marginal significance if it is strongly related to the query and is least different from the earlier document. MMR helps to find out the relevant candidate documents quickly and to find the similarity between them. If the summary is to be found via relevant sentence extraction, it requires relevance and redundancy to be discovered out. In single document summarization, the documents are divided into sentences, cosine similarity is found out, and sentences are ranked for the summary. In MMR, the candidate selection score has two components; one is the relevance of the candidate with the user’s query, and the other is a similarity of a selection of candidates with other candidates present in summary. These scores are computed in each iteration, and the algorithm stops after meeting specific criteria. MMR method works well for long documents as they have more repeated sentences. It is also suitable for the extraction of sentences about a similar topic in multi-documents. It helps in eliminating redundancy in query-relevant multi-document summarization. The problem with this algorithm is it does not help in reducing the global diversity, and it does not provide the facility scale to output with a larger size.

Lin et al. [57] characterize the interactive summarization technique by using the MMR algorithm, which helps the user select candidate sentences. This helps in generating highly interactive high-quality summaries than automatic summaries. Lin et al. [57] extended MMR [56] algorithm, which places users in a loop. The user is asked to select a sentence at each step that would be added to the summary. It gives the user a ranked list of sentences for selection. The evaluation vehicle for measuring the summarization algorithm’s effectiveness is Complex, Interactive Question Answering (CiQA). CiQA consists of topics that have two parts, i.e., question template and narrative (description). Participants organize web-based QA systems with which NIST assessors interact. Each assessor interacts with the participant, after which participants submit the final run. In experimentation, interactive MMR is executed after the initial run (standard run) is performed.

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

TABLE II
STRENGTHS AND WEAKNESSES OF CLUSTERING-BASED METHODS

Sr.#	Research Study	Working	Results & Evaluation
1	Zhang et al., 2015	Representativeness, Diversity, and Length parameters are considered in the clustering-based method of MDS.	Performance at DUC 2004 is good. However, it does not handle synonymy and polysemy problems. Work is still in a preliminary stage, and refinements are in progress.
2	B. Wang et al., 2017	It eliminates the need to tell in advance the number of desired clusters due to Density Peaks' use.	ROUGE 1 and SU4 give best results. However, it does not handle synonymy and polysemy problems which gives lots of future directions for researchers.
3	Nagwani, 2015	Summarization of extensive data available in BigData is performed using the MapReduce framework.	The algorithm designed in the study is evaluated on some legal documents. It would give us a better understanding of results if implemented on benchmark datasets. Similarly, the technique uses external resources extensively, which makes it an expensive MDS solution.
4	Christensen et al., 2014	Hierarchical summarization is presented where nodes provide additional information if we keep traversing until the leaf node is reached.	Redundancy and budget are treated as hard constraints and coherence and salience as soft constraints.
5	Rodriguez & Laio, 2014	Cluster borders are managed by calculating local density and high-density points.	In some cases, the datasets with a small number of points are affected by significant statistical errors.
6	Contractor et al., 2012	Sentences are labeled based on whether they appear in the abstract, main body, etc., of articles using Argumentative Zoning.	Clustering and classification are used together, which increases the cost. Positive and negative labeling of sentences is complex.

In interactive MMR, which is web-based, the user selects sentences at every step. For the final run, the output of the interactive run is combined with the output generated automatically. IDF is used to compute the relevance of each document in the experiment. Cosine similarity is used to eliminate redundancy. The interface consists of 3 components; question, current answer, and sentences ordered as scored by MMR. At each step, the user is asked to select the sentence which is then added into the current answer. F-measure is considered for evaluation measure. But the problem is it does not account for the sentences which have varying length. The weighted answer shows that how far relevant information is contained in the system response. Another downside of the solution is the human intervention which is necessary for the task but is hard and time-consuming.

Ozsoy et al. [58] tried to solve the shortcomings of previous approaches. The earlier methods first select the concept and then choose the sentences related to the concept, which is finally used in summary. Ozsoy et al. [58] used LSA-based methods on Turkish text and devised two techniques of sentence selection. The Cross method was used for sentence selection in the input matrix. This method's primary function was to determine that although sentences at the introduction and conclusion part tend to be more critical, there can still be some sentences selected that may cause noise in the matrices of LSA. Like previous approaches, the Vector Transpose VT matrix is used. The Cross method pre-processed this matrix before sentence selection. The average sentence score was calculated for each concept in VT matrix for every row. For cell values less than the average row score, they were set to zero, for these were sentences related to a topic somehow but not the core sentences. Then the length score is calculated for sentences. The sentences are selected based on higher values. To distinguish between the main topic and the subtopic, another method, named Topic method was proposed. It decided the main topic by creating a concept * concept matrix. This matrix added the cell values that were common among concepts. The strength value of concepts was calculated by considering each concept as a node and the similarity value of concepts * concept as edge score. Then values of concept in each row of this matrix are added to compute the concept's strength. Higher value concepts are

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112496, IEEE Access

considered as the main topics. Investigation on two data sets was performed, which was then related to human-generated abstract summaries. The concern here was the use of complex algorithms with SVD.

Data representation is complex in textual data, as it suffers different problems like uncertainty, imprecision, incompleteness, etc. This causes the problem of classifying the same sentences into different classes. This paper [59] uses Fuzzy Rough Sets (FRS henceforth) based sentence similarity measures because FRS uses meanings of sentences. FRS is the combination of Fuzzy Sets and Rough Sets. Former deals with uncertainty through membership functions, while the latter with the help of lower and upper approximation of a set. Imprecision can be defined as something that is not precisely told. For example, consider the sentence “Ram is a man of medium height,” We have no idea about what a medium height stands for. On the other hand, uncertainty occurs due to polysemous words, anaphoretic pronouns, and structural ambiguity.

The lower and upper approximation is estimated as those that certainly belong to the concept make its lower approximation. In contrast, the elements that possibly can belong to the concept make an upper approximation. The technique for sentence similarity was tested on the SICK 2014 dataset, while for summarization DUC 2002 was used. Results reported on DUC 2002 were quite encouraging for ROGUE 1, ROGUE L, and ROGUE SU.

Single document summarization is used for the extractive method [60]. The technique adopted is CNN, and it is tested on the datasets of CNN, Dailymail, and NYT. They compare the extracted sentences, keeping a particular focus on the grammar quality of the resultant sentences. Sentences are encoded using bidirectional Long Short-Term Memory (LSTM) and the Convolutional Neural Networks (CNN). Sentence representatives are identified and then aggregated with document representative that is encoded with bidirectional LSTM and CNN. Decoding is done with sequential LSTM. Greedy decoding is then applied at the testing phase for the nomination of the most likely sentence sequence. These selected sentences are then compressed by omitting some words or phrases to make them more concise. Compression rules and feed-forward networks facilitate the choice of deletion of words or phrases. The resultant summaries are evaluated at mTurk, Grammarly, and manual analysis. With the CNN dataset, the results were more promising as it contains compressed sentences already.

The methods based on LSA for MDS are presented in Table III for review.

3) NON-NEGATIVE MATRIX FACTORIZATION (NMF) METHODS

In non-negative matrix factorization-based methods, factorization is performed on the sentence-term matrix to determine the highest probability sentences within each topic. It is more like a clustering technique with all its benefits [37], [38], [61]. Sentences are clustered as per their set criteria, and salient sentences within clusters are then determined and summed up to create the summary.

C. RHETORIC STRUCTURE THEORY-BASED METHODS (RST)

Rhetoric Structure Theory, or RST based methods, as depicted in Figure 3, divide the text into adjacent textual units that are consecutive sentences and apply different RST rules on text units to see each unit’s importance. It ranks the sentences into nuclei and satellites, where nuclei are the important sentences that need to be included in the summary, and satellites contain additional information about nuclei. RST based methods are also considered in MDS [43].

Automatic Summary generation might result in poor grammatical quality. This problem is dealt with in work by Durrett et al. [43] in which Anaphoricity constraints are considered while compressing the sentences for summarization. It divides the text into text units, performs compression by Rhetoric Structure Theory by further dividing the sentence into Elementary Discourse Units (EDU), and Syntactic Compression is then applied so that the given sentence is easily compressed by considering the noun phrases, pronoun phrases, and other RST based rules while selecting the EDUs like elaboration statements for deletion. It also uses the pronoun replacement to remove any ambiguity and inconsistency from the summary. A situation arises when the statement with a pronoun is included in the summary while it’s antecedent (the statement containing the actual proper noun or simply the noun) is omitted from inclusion. The system then replaces the pronoun in two possible ways. It either picks the noun from the antecedent statement. It replaces it with the pronoun used in the selected statement, or in case the replacement is not that straightforward, it includes the entire antecedent statement in summary. Supervised learning is done through the structured SVM technique. This algorithm, however, worked for single-document summarization.

D. GRAPH-BASED METHODS

As presented in Figure 4, graph-based methods construct graphs of sentences that are part of the document collection. The sentences make the graph’s nodes, and edges are either drawn based on the similarity between sentences fulfilling the threshold criteria or belongingness to the same document. Voting of neighboring nodes selects sentences to generate a summary. Erkan and Radev [31] devised the LexPageRank algorithm based on eigenvector centrality (prestige) to determine significant sentences, as was done successfully in the Google PageRank algorithm.

Ercan Canhasi [62] presented a technique based on Five-Layered Heterogeneous Graph and Universal Paraphrastic Embeddings for query-focused extractive multi-document summarization. In this work, the focus is on sentence and document level relations and includes part of sentence similarity and query to sentence similarity.

TABLE III
STRENGTHS AND WEAKNESSES OF LSA-BASED METHODS

Sr.#	Research Study	Working	Results & Evaluation
1	Gong & Liu, 2001	SVD was used to derive the Latent Semantic structure. Weights are assigned to sentences, and those with more weights are selected in summary.	It is one of the earlier studies in this field, so it would be best to test it on benchmark datasets. Moreover, slight disparities in sentence selection are observed, which increases with the length of the documents.
2	Ferreira et al., 2016	Sentence similarity, word order, and meaning problems are handled in a 3-layered module of lexical, syntactic, and semantic layers.	It would be best to test on the standard dataset like DUC. The performance of semantic and syntactic measures showed promising results when the lexical layer was added. These measures can be improved on an individual level without a lexical layer.
3	Marujo et al., 2016	KPCentrality method of SDS is extended here for MDS. It works as a single layer as well as a waterfall approach.	The results show an improvement of 16% at ROUGE-1 scores for TAC 2009 and 17% for DUC 2007. Researchers' need to work on the area as intermediate summaries do not include all important events.
4	Carbonell & Goldstein, 1998	MMR approach considers the relevance of sentences with query and other sentences in the documents.	As long as the topic remains the same, the results are promising. It doesn't work well to extract sentences from multiple topics in documents and offers a fertile field for a researcher for improvements.
5	Lin et al., 2010	This technique provides interactive, high-quality summary generation using MMR that keeps the user in a loop during processing.	Since the user is on-board during the summarization process, it is effective, but at the same time, it causes delays due to human interactions, thus, is time-consuming. Sentences with varying lengths are not handled.
6	Ozsoy et al., 2010	The Cross method is presented to handle the problem of noisy sentences selected for summary, causing the error. The topic method is used to identify the main\subtopics of sentences.	A simple and effective technique of summarization. Currently, it is tested on different scientific article of Turkish language. If tested on standard datasets, it will be helpful in the research for better comparisons.
7	Chatterjee & Yadav, 2019	The Fuzzy Rough Set method is used to deal with sentence similarity and uncertainty problems within data.	Results reported on DUC 2002 were quite encouraging for ROGUE 1, ROGUE L, and ROGUE SU, while for the SICK 2014, the improvements can be made to get better results
8	J. Xu & Durrett, 2020	CNN and LSTM are used on extractive single-document summarization.	The resultant summaries are evaluated at mTurk, Grammarly, and manual analysis. With the CNN dataset, the results were more promising. It is recommended to apply compression over NYT and Dailymail datasets in order to get a better result there as well.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112496, IEEE Access

Sentences are iteratively ranked using the PageRank [30] algorithm. To calculate the text similarity, universal paraphrase embeddings are used. The technique in this paper was implemented on benchmark dataset DUC 2005. The performance was evaluated on ROUGE 1 and ROUGE 2 as next to reference summary, while on ROUGE SU4, their performance deteriorated.

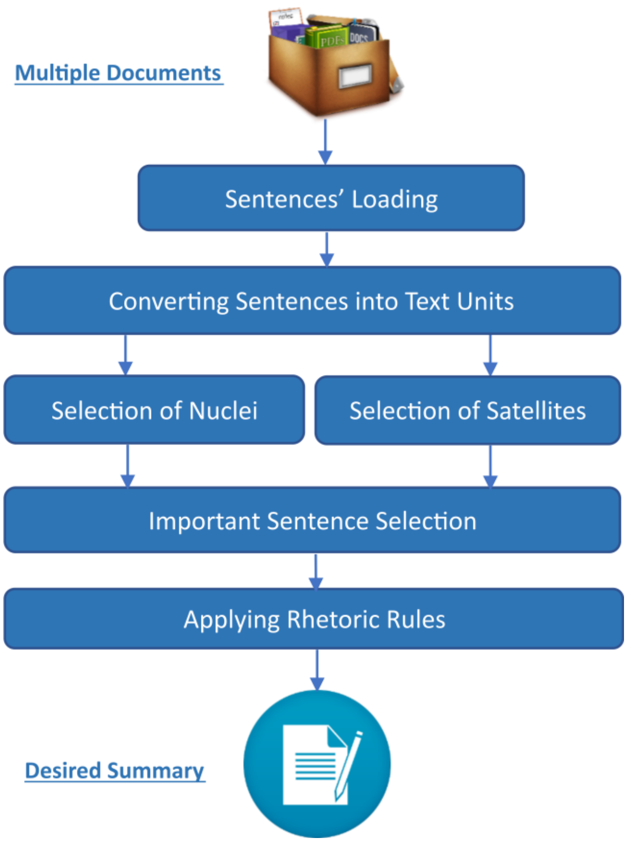

FIGURE 3: **RST-based multi-document summarization**

Shafiei et al. [63] presented a word graph-based method for the multi-sentence compression (MSC) approach. They used substantial merging, mapping, and re-ranking modules that resulted in more compressed summaries by retaining informative and grammatically sound sentences. Multiword Expressions (MWE) are handled by substituting an MWE with its one-word synonym and make it a node in the graph. This removes ambiguity and results in compression as well. It handles the concept of synonymy by replacing the up-coming one-word with its already existing synonym node in the graph. It uses a 7-gram POS-based language model (POS-LM) to rank the k-shortest paths obtained from the graph without compromising the resulting compressed sentence's grammar. It can be said safely that this is the first time to use MWEs, synonymy, and POS-LM for improvement in the quality of word graph-based multi-sentence compression. This approach is tested extensively on the standard datasets and has shown effective results for compression with grammaticality.

Multi-document text summarization has also experimented with data mining techniques. Baralis et al. [64] applied Association Rule Mining of data mining to see the results of its over summarization process. They devised the GRAPHSUM algorithm to find out correlations between multiple terms in graph-based summarization. Apriori algorithm was adopted to do association rule mining to find correlation among terms, and then PageRank [30] was used to rank salient sentences.

Graph techniques are also effective in many other problem-solving methods. For instance, Chali et al. [65] presented a system for answering complex questions by the random walk method of graph-based technique and measured the effect of syntactic and semantic information in it. They measured the similarity among sentences by applying tree kernel functions in the random walk framework. Then, they extended the work further to incorporate the Extended String Subsequence Kernel (ESSK) to perform the task equivalently.

Vertex Cover algorithm-based multi-document summarization was presented by John et al. [66] using sentences' information content. The vertex cover algorithm worked like the famous Euler's graphs. To cover all edges, a graph was constructed where vertices were a subset of the original graph. Vertices represented sentences, and edge scores represented relevance with other sentences. Vertex (which was a sentence) with a higher relevance score would appear in the final graph, i.e., summary in this case.

Archetypal analysis is an unsupervised learning technique that works in the same manner as cluster analysis. Archetypes are the external points in multidimensional data, and that is how they differ from typical observations like cluster centers. Archetypal analysis was used to check for any improvement in a query-focused MDS [21], [67] with weighted element graphs and hierarchies.

Tzouridis et al. [68] stated that connected sentences could be represented by using the word graph so that the shortest path makes up summaries. They used parameterized shortest path algorithm and the large margin approach for sentence compression. This approach is superior to other multi-sentence compression approaches. They used the structured approach of learning in multiple sentence compression. Parameters are adjusted in the shortest path algorithm. Data labelling is done through a structured expectation framework. Features are used to embed the word graph and its shortest paths which consequently become the desired summaries. The linear scoring function learns to differentiate between the different quality of compressions. The integer linear program is used to solve the problem that works in polynomial time. Related sentences become input to word graphs. Unique words of sentences become vertices of the graph and directed edges that connect words of at least a sentence. A path in the graph is the connected sentence. It extends the work to a structured prediction framework using parameterized shortest

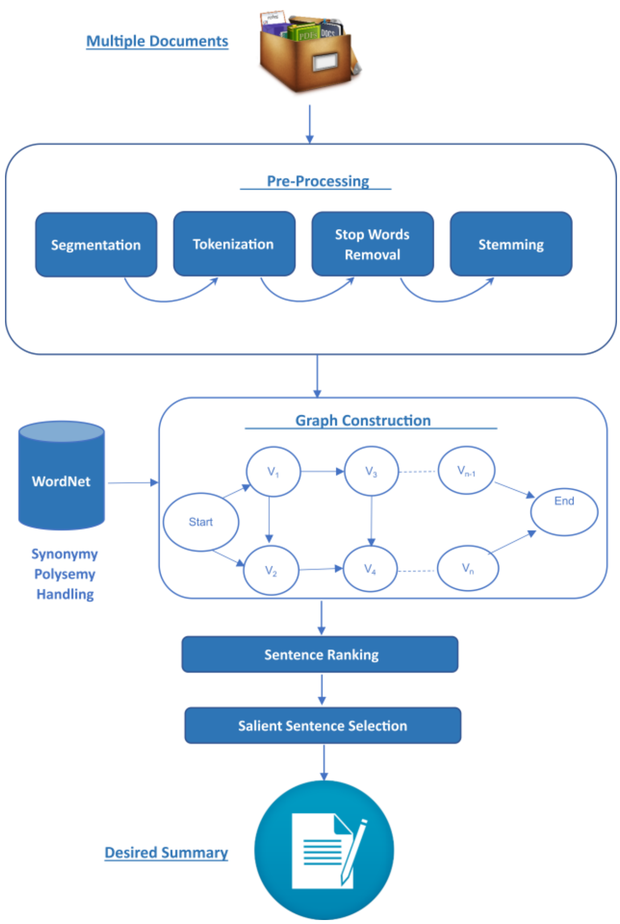

FIGURE 4: **Graph-based multi-document summarization**

path algorithm. It uses SVM for the shortest path algorithm to learn the shortest path for a highly dimensional feature and proposes a polynomial-time procedure. It also uses the shortest path for the experiment of significant news. The edge weights are used based on word frequency. Some scientists use the key phrase method to generate summaries. The words used for the graph must be pre-processed Sometimes, complex pre-processing is required, such as reunion vertices containing replacements. The shortest path algorithm figures the cost by adding all the edges in the path, such the path p has vertices between Start and End. To summarize related sentences, it is needed to find the function that gives the best summary and assign the minimum score to the best summary. To assess function f, a hamming function is used. Then the task is to find the position function that gives the smallest score to the best summary. After that margin-rescaling technique is used. The margin method is used to fetch the margin among the best path and all other paths. Decoding of P^ is used for margin scaling to scale the margin with the real loss. The margin technique also affects the central loss, which is greater than structural loss. Experiments were done on a set of a predefined set of categories i.e., news about sports business, etc., and the pre-processing was done by using spectral clustering. A fully connected graph was created in which vertices were headlines, edges were weighted by the number of shared non-stopwords. After that clustering was achieved day by day, and the resulting data was considered as headline news about the event. The data with high probability was measured as the data about the occurrence, and it was the related input sentences. For best summary identification, crowdsourcing is used. The annotator has given n number of sentences and must create ten summaries using Yen's algorithm. After that, the best summary is marked. Then three most appropriate summaries are collected. The learning approach uses the following method: Every edge is associated with a feature vector. The feature vector consists of the join frequency, maximal word frequency, lexical relevance, normalized Pointwise Mutual Information (PMI), the average location of the phrase. The experiment uses the holdout method with a distinct holdout method and test sets. Analysis reveals that the positive correlation of graph density is connected to a negative correlation of lexical diversity.

In the technique devised in [69] graph-based method is used in which nodes are represented by sentences and edges characterize the preference value of the sentence. It uses the entailment method in which one sentence's meaning can achieve the meaning of an alternative sentence. This entailment can be found by some symmetric and non-symmetric measures. In pre-processing unit, tokenization is performed, and stop words are removed. The significance of the word in the similarity matrix is calculated through the tf-idf and weight. After that, sentence ordering is achieved based on preference measures that comprise topical closeness, chronology, precedence, succedence, semantic, and text entailment experts. This system deals with the semantic relationship, rational conclusion so that the meaningful summary is generated and emphasizes evidence extraction and sentence order. WordNet is used for the semantic link between the sentences, which creates the rational entailment between the summary sentences. The primary module is text entailment expert that investigates the logical relationship among the sentences by using symmetric and non-symmetric measures. The symmetric measure is calculated by using a cosine measure to find the similarity statistically. The additional module is the ordering of sentences. The sentences are extracted from the documents, and the total preference value is calculated. After that, the ordering algorithm will perform the ordering in the following way. Experimental results show that the entailment method for sentence ordering and ranking provides high precision and provides a well-organized summary that significantly helps the reader realize data. This technique, however, does not focus on coherent summaries. Similarly, the use of non-symmetric similarity measures and complex algorithms make it a bit costly solution.

The technique in [70] focused on G-FLOW is a novel method using the joint model. The work focuses on the technique used to resolve the problem of selecting sentences along with the sentence ordering problem. It constructs the directed graph, where sentence represents vertex and connection between the sentences si and sj means that sj can be placed right after si in summary. Need is to identify sentences that have the relationship among them. This method first automatically constructs the graph for multi-document summarization, which requires innovative methods for identifying inter-document connections. It then uses this graph to find the coherence of the specific sentence. After that, G-FLOW uses a technique for sentence collection and order. Previous procedures did not emphasize coherence between sentences and selected disconnected sentences. This technique generates summaries without any domain-specific knowledge and identifies coherent documents rather than sentences. The aim is to develop a pair-wise ordering constraint and which specifies a discourse graph, which is then used by the G-FLOW graph to estimate coherence. Textual cues are from literature, and the redundancy naturally presents in connected documents used to produce edges. The technique focuses on generating coherent summaries based on jointly improving coherence and salience. It generates a summary using ADG (approximate discourse graph) where each node is the sentence and edges represent the discourse relationship. Experimentation demonstrations give better results than other MDS techniques. The matter is coherence and salience are less focused. WordNet is used, so more training is required.

In the graph-based methods. The technique by John and Wilscy [66] gives best results on DUC 2002 with Rouge-2 values of 0.07059, whereas in DUC 2007, Chali et al. [65] came up with Precision value of 0.392012 in Rouge-1.

Table IV presents the gist of methods working on the graph-based technique.

D. MISCELLENEOUS METHODS

The term-based multi-document summarization fails to handle synonymy and polysemy problems, while ontology-based summarization can work well only where the ontologies are already defined. The definition of ontology involves a great deal of workforce to define it. To overcome both the concerns, Qiang et al. [16] came up with a closed pattern-based technique for MDS, which extracts the important sentences from document collection using closed patterns to decrease repetition in summary. Their method, PatSum, calculates the sentence weight in the document(s) by adding the weights of its covering closed patterns concerning the current sentence and repeatedly selecting a sentence with the highest weight and less similarity to the previously selected sentences, until the length limit is reached. This technique reduces the dimension while retaining the related information. PatSum method uses the advantages offered by the term-based and ontology-based methods without adopting their weaknesses. Extensive experiments on the benchmark DUC2004 datasets show that the pattern-based method outperforms the state-of-the-art methods significantly.

Evolutionary algorithms are used to optimize the search space. In work proposed by Rautray and Balabantaray [71], Cuckoo Search (CS) algorithm is applied as a solution to the generic extractive multi-document summarization problem. The authors have compared their technique with two other evolutionary algorithms named Particle Swarm Optimization (PSO) and Cat Swarm Optimization based (CSO) summarizers. They have found out that CS-based summarizer results are better on the benchmark datasets of DUC 2006 and DUC 2007. However, since CS belongs to the evolutionary algorithms, they have a problem with controlling parameters. Therefore, this was also faced in the implementation of CS in generating summaries in MDS.

In this paper [72], the authors used the Bat Algorithm of optimization to the objective function in search of the optimal solution. At the start, the data is divided into sentences, which consequently are divided into words. Then pre-processing is applied by removing stop words and converting the data into lower case. The objective function is designed to address two objectives:

a) It should give proper coverage

b) Redundancy should be avoided in summary sentences

Indian dataset is used to test the technique. Indian dataset contains 4516 news articles along-with the gold standard summaries. For the evaluation, ROGUE 1, ROGUE 2 are used, and the comparison of the summary of their technique was made with the summary generated by MS Word.

The three most essential points the best summary must contain are coverage, non-redundancy, and relevance. To achieve such a summary, the authors [73] used Shark Smell Optimization (SSO) for multi-document summarization. SSO uses the word embedding-based similarity function and Google-based similarity function, and SSO calculates optimal weights of text features. Word Mover's Distance is a word embedding technique-based distance function to find the similarity among the text documents so that the embedded words of the first document need to travel to reach the embedded words of the second document. In contrast, Normalized Google Distance is a Google hit-based dissimilarity function.

The technique was tested on DUC 2004, DUC 2006, DUC 2007, TAC 2008, TAC 2011, and MultiLing 13.

The authors in [74] used textual entailment relations and sentence compression by the Knapsack problem. It is used to address the extractive MDS problem.

It first ranks the sentences by tf-idf method and then calculates the entailment scores of the selected sentences. The sentence's final score is calculated, and then the sentences are compressed through a greedy dynamic programming approach for the Knapsack problem. This technique gives 2% improvement in the query-based approach of summarization, while for the generic summary, 5% improvement is recorded.

The knapsack problem is one of the optimization problems. Here, sentences are considered problem items, and their values are calculated by "production" of entailment score and tf-idf value. ROGUE 1, ROGUE 2, ROGUE SU4 are used for evaluation, while the datasets selected were DUC 2007 for query based and MultiLingPilot 2011 for generic summarization. In this paper [75], the authors devised three methods for sentence selection, namely sentence-context relevance, sentence novelty, and sentence position relevance for the methodology SummCoder for a summary generation. These sentence features are fused to rank and select sentences for a summary of the given length. TIDSum dataset is used to test the methodology, along-with DUC 2002, and Blog Summarization Corpus. Unsupervised deep auto-encoder was trained such that Recurrent Neural Networks (RNNs) encoder with Gated Recurrent Units (GRUs) and RNN decoder with conditional GRUs.

TABLE IV
STRENGTHS AND WEAKNESSES OF GRAPH-BASED METHODS

Sr.#	Research Study	Working	Results & Evaluation
1	Canhasi, 2017	5-layered heterogeneous graph method is presented that also handles paraphrases.	It outperforms the other baseline implementations. However, the performance is not up-to-the-mark in ROUGE-SU4.
2	ShafieiBavani et al., 2016	A language model has been used in a word-graph-based sentence compression technique that replaces the MWEs with its one-word substitute along-with the contemporary synonym replacement.	Grammaticality is increased. It would be better to focus on the informativity of the selected summary.
3	Baralis, Cagliero, Mahoto, et al., 2013	Apriori algorithm of association rules is used with a graph-based technique, named as GraphSum, to find a correlation between terms.	This technique employs the Apriori algorithm, which scans the dataset many times. It is considered to stay in primary storage mostly and is an expensive solution based on time and space complexity, so efforts are needed to improve the efficiency.
4	Chali et al., 2011	The random walk method of the graph is used for the complex question answering system	Repeating entities, in summary, are not considered for dereferencing.
5	John & Wilscy, 2015	Vertex cover algorithm is presented for text summarization that must cover all the edges in the graph. Relevance of edge defines sentence salience for inclusion in prospected summary.	Overall results are promising. Sentence relevance of the selected ones for summary needs improvement.
6	Tzouridis et al., 2014	Word-graph-based compression among sentences is used with supervised learning using SVM.	Word graph technique and SVM-based learning make the compression better. However, a limited feature set is used, which can be tested with an enhanced feature set.
7	Sukumar & Gayathri, 2014	Semantic relationship and rationale are the main focus of this study by emphasizing evidence extraction and sentence ordering using the sentence entailment method.	It gives better results in sentence extraction and ordering. However, it does not focus on the coherence of summaries.
8	Christensen et al., 2013	Deverbal noun method is presented for sentence selection and order problems.	With increased training, better results are expected.

IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

ROGUE recall factor for R1, R2, Rogue L, ROGUE SU4 are applied.

There are different other methods like CRF-based summarization and Hidden Markov Model (HMM) based method. Table V shows the pros and cons of miscellaneous methods working in MSD.

E. SECONDARY STUDIES CONDUCTED IN MDS

MDS has attracted many authors for performing secondary studies as well.

This paper [76] briefly discusses the different techniques of extractive and abstractive summarization. They explored the different pros and cons of both types of summarizations and proposed that a mixed approach should be used for better summary generation.

A detailed survey [77] is conducted to investigate the focus of current studies in text summarization. The authors also helped the new researchers by projecting the research gap in this field. A similar survey was conducted by [78] on legal documents. The study investigated the text summarization methods devised for the legal documents' summarization and collected the performance comparisons of different techniques and the different datasets for the interested researchers.

TABLE V
STRENGTHS AND WEAKNESSES OF MISCELLANEOUS METHODS

Sr.#	Research Study	Working	Results & Evaluation
1	Qiang et al., 2016	Closed patterns are applied to find the shortest path for MDS. The solution, named PatSum, is compared with ontology and term-based methods.	For larger support value, the performance of PatSum declines.
2	Rautray & Balabantaray, 2018	An Evolutionary algorithm, called Cuckoo Search, is applied in MDS.	The parameter controlling problem of evolutionary algorithms needs to be resolved.
3	Anshuman Pattanaik, Santwana Sagnika & Mishra, 2019	Bat algorithm for optimization is used to search the optimal solution, to maximize the coverage and minimize the repetition	Can be tested on DUC, TAC, and other benchmark datasets
4	Verma & Om, 2019	MCRMR algorithm is designed by using the Shark Smell Optimization technique on MDS for best results	With Machine Learning based methods, the results can become better.
5	Naserasadi, A., Khosravi, H., & Sadeghi, F. (2019).	Sentences are ranked, and then entailment scores are calculated and then finally compressed using 0-1 Knapsack problem.	Gives 2% improvement in the query-based approach, while 5% improvement is recorded for the generic summary. Efforts are required to decrease the complexity of the algorithm
6	Joshi et al., 2019	SummCoder technique is proposed, which comprises sentence-context relevance, sentence novelty, and sentence position relevance.	It gives promising results on single-document summarization. Can be extended on MDS

In another secondary study [79], a systematic literature review was conducted to investigate the status of importance and significance of fuzzy logic in text summarization. They designed the research questions to conduct this study on electronic research databases, like, IEEEXplore, ACM Digital Library, ScienceDirect, GoogleScholar, Springer, and Wiley Digital Online. After performing the respective inclusion-exclusion, 52 articles qualified to be included in this SLR. 49 were primary studies, and 3 were secondary studies on fuzzy logic for text summarization. Further quality assessments finally resulted in 42 total studies in SLR, 39 were primary studies, and 3 were secondary studies. The findings of SLR affirmed the importance and emerging trend of the use of fuzzy logic in text summarization.

III. DATASETS

DUC-Document Understanding Conference: Since 2001, the Document Understanding Conferences is playing the role of an effective forum for researchers in automatic text summarization to compare common test sets' methods and results. They release datasets having benchmark document collections from multiple sources on an almost yearly basis. It also includes the human-generated reference summaries so that users may compare their candidate summaries (generated by the individual algorithms) with them [9], [10], [55]. Majority authors [1] – [6], [9] – [12], [14], [16] – [21], [24], [25], [32], [35], [41], [48], [51], [55], [64], [65], [66], [67], [71], [80], have used DUC to observe the performance of their technique.

TAC- Text Analysis Conference: Like DUC, TAC is also a collection of benchmarked documents from multiple sources, accompanied by human expert-generated summaries for reference. The difference is that TAC is extended with support for other languages [8], [12], [15]. The authors [15], [18], [55], [67] tested their techniques on TAC.

The other datasets used are as following:

TSC-3 – (Text Summarization Challenge corpus) is used by [44], [45]. Similarly, RSS Feeds, New York Times annotated corpus, TREC 2007 are used apart from the user-generated datasets.

IV. EVALUATION TECHNIQUES

ROUGE - Recall-Oriented Understudy for Gisting Evaluation: It is a set of metrics and a software package used for evaluating automatic summarization and machine translation software in natural language processing. The metrics compare an automatically produced summary or translation against a reference or a set of references (human-produced) summary or translation. Authors in [2] – [6], [8] – [12], [14] – [21], [24], [25], [29] [35], [39], [41], [43], [48], [51], [54], [55], [62]–[68] , [70], [71], [80] used ROUGE for the validation of their results.

BLEU- Bilingual Evaluation Understudy: It is a special algorithm for quality evaluation of machine-translated text between natural languages. It evaluates the translation done by machine with its closeness with human translation on the measure of fluency and adequacy. It is used in [63], [68], [70].

Other evaluation metrics used are Precision, Recall, F-measure, Average Continuity, Pyramid, SemEval, Correlation Coefficients, Amazon mTurk, etc.

V. DISCUSSION

This article presented a recent survey of previous work on using extractive techniques for multi-document summarization. It would provide a perfect starting point for the researchers to contribute to the field of multi-document summarization. Extractive techniques can be divided into Term-based methods, Rhetoric Structure Theory-based methods, graph-based methods, and several other variations of the most standard methods. The working of these techniques are individually explained, and then a thorough discussion on the important studies conducted is carried out. We discussed the pros and cons of each method under different training conditions. We also presented the most commonly available datasets that are used to evaluate and compare new summarization techniques. In the end, we presented the evaluation matrices. By mentioning the strengths and weaknesses of the discussed techniques, we have tried our best to point out the different future directions for the newcomers in this field of research to focus their study on. Table 1-5 can be especially beneficial for the readers who wish to find research problems to kick start their research process. Different studies can be considered for improvements. For example, the techniques of [4], [36], [45], [50], [58], [72] can be rigorously tested on dataset like DUC, TAC, NYT, and other benchmark datasets, discussed in section 3.

Similarly, the work of [46] can show significant improvements if pre-processing is also enhanced with the step of stemming. It is worth mentioning that, even though polysemy and synonymy are iterated to be reported as an open problem in the field of MDS in literature, it has not attracted much attention of the researchers so far. The interested researchers can kick start their endeavor by handling it in many studies, for instance, [48], [49].

One of the observations of this study is that external resources like WordNet are used for synonym mapping extensively. All such studies can be replicated by using word embedding techniques like Word2Vec, GloVe, etc., for the synonym mapping. Moreover, in [55], all important events were not included in the intermediate summary. In [81], the sentence extraction from multiple topics did not work well, while in [57], the progress deteriorates with varying lengths of summaries. While [63] improved their summary's grammaticality to a satisfactory extent, they could not make it for the informativity, which offers a fertile field for the upcoming researchers' improvements. The feature-set of work of [68] can be extended for promising results; on the other hand, [69] offers the researchers to work on the improvement of summary cohesion. In the end, the summarizers working well in SDS [43], [75] can be tested for improved performance in MDS as well.

VI. CONCLUSION

Automatic Text Summarization systems are increasingly gaining the interest of the users to obtaining the concise version of the lengthy and redundant textual documents, without skipping any important piece of information. In this survey paper, we aimed to gather the state-of-the-art techniques published in different studies over the last decade about extractive multi-document summarization. In this survey, we discussed in detail the various techniques of extractive MDS, like i) ontology-based methods, ii) term-based methods (that can further be classified into clustering methods, latent-semantic methods, and non-negative matrix factorization), iii) rhetoric structure theory-based methods, and iv) the graph-based methods. We proposed and discussed the different guidelines to facilitate the new researchers in this field to make a start, with emphasis on the abovementioned techniques. While discussing different studies, we have analyzed them critically and pointed out the pros and cons. In section 5, different open problems can be considered by the research community for further improvements. The open problems are as follows:

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

i) Diversity: To increase the diversity of the summary. Every topic that is mentioned in the document clusters, should be mentioned in the summary, it must not be focused upon just one of the many topics found in the document clusters.

ii) Redundancy: the text summarization systems mainly suffer from the repetition of the same fragments of information in the summary, ignoring many important points, therefore. The need is to devise a summary in such a manner that repetition should be minimized, if not eliminated.

iii) Informativity: the summary must be carrying the information in a precise and concise manner. Extractive summarization involves extracting the fractions from the given documents; therefore, it mainly suffers from the lack of informativity concerns. An effective summarizer must convey the information in a compact way to the reader.

iv) Grammaticality: quality of the summary suffers from grammar due to connecting the extract of the different chunks from the document set. The need is to make such a system that refines the summary for grammar at the end.

v) Urdu is among the popular languages spoken in the world, but unfortunately, no summarizer is made to handle it in its script. It will be a very good combination of language and image processing research areas.

REFERENCES

[1] R. Ferreira et al., “A multi-document summarization system based on statistics and linguistic treatment,” Expert Syst. Appl., vol. 41, no. 13, pp. 5780–5787, 2014, doi: 10.1016/j.eswa.2014.03.023.

[2] A. Khan, N. Salim, and Y. Jaya Kumar, “A framework for multi-document abstractive summarization based on semantic role labelling,” Appl. Soft Comput. J., vol. 30, pp. 737–747, 2015, doi: 10.1016/j.asoc.2015.01.070.

[3] G. Yang, D. Wen, Kinshuk, N. S. Chen, and E. Sutinen, “A novel contextual topic model for multi-document summarization,” Expert Syst. Appl., vol. 42, no. 3, pp. 1340–1352, 2015, doi: 10.1016/j.eswa.2014.09.015.

[4] H. Oliveira et al., “Assessing shallow sentence scoring techniques and combinations for single and multi-document summarization,” Expert Syst. Appl., vol. 65, pp. 68–86, 2016, doi: 10.1016/j.eswa.2016.08.030.

[5] G. Glavaš and J. Šnajder, “Event graphs for information retrieval and multi-document summarization,” Expert Syst. Appl., vol. 41, no. 15, pp. 6904–6916, 2014, doi: 10.1016/j.eswa.2014.04.004.

[6] Y. J. Kumar, N. Salim, A. Abuobieda, and A. T. Albaham, “Multi document summarization based on news components using fuzzy cross-document relations,” Appl. Soft Comput. J., vol. 21, pp. 265–279, 2014, doi: 10.1016/j.asoc.2014.03.041.

[7] E. Barbu, M. T. Martín-Valdivia, E. Martínez-Cámara, and L. A. Ureña-López, “Language technologies applied to document simplification for helping autistic people,” Expert Syst. Appl., vol. 42, no. 12, pp. 5076–5086, 2015, doi: 10.1016/j.eswa.2015.02.044.

[8] H. Oufaida, O. Nouali, and P. Blache, “Minimum redundancy and maximum relevance for single and multi-document Arabic text summarization,” J. King Saud Univ. - Comput. Inf. Sci., vol. 26, no. 4, pp. 450–461, 2014, doi: 10.1016/j.jksuci.2014.06.008.

[9] R. M. Alguliev, R. M. Aliguliyev, and M. S. Hajirahimova, “GenDocSum + MCLR: Generic document summarization based on maximum coverage and less redundancy,” Expert Syst. Appl., vol. 39, no. 16, pp. 12460–12473, 2012, doi: 10.1016/j.eswa.2012.04.067.

[10] W. Luo, F. Zhuang, Q. He, and Z. Shi, “Exploiting relevance, coverage, and novelty for query-focused multi-document summarization,” Knowledge-Based Syst., vol. 46, pp. 33–42, 2013, doi: 10.1016/j.knosys.2013.02.015.

[11] R. M. Alguliev, R. M. Aliguliyev, and N. R. Isazade, “Multiple documents summarization based on evolutionary optimization algorithm,” Expert Syst. Appl., vol. 40, no. 5, pp. 1675–1689, 2013, doi: 10.1016/j.eswa.2012.09.014.

[12] R. M. Alguliev, R. M. Aliguliyev, and N. R. Isazade, “DESAMC+DocSum: Differential evolution with self-adaptive mutation and crossover parameters for multi-document summarization,” Knowledge-Based Syst., vol. 36, pp. 21–38, 2012, doi: 10.1016/j.knosys.2012.05.017.

[13] C. Sunitha, A. Jaya, and A. Ganesh, “A Study on Abstractive Summarization Techniques in Indian Languages,” Procedia Comput. Sci., vol. 87, pp. 25–31, 2016, doi: 10.1016/j.procs.2016.05.121.

[14] R. M. Alguliev, R. M. Aliguliyev, and C. A. Mehdiyev, “Sentence selection for generic document summarization using an adaptive differential evolution algorithm,” Swarm Evol. Comput., vol. 1, no. 4, pp. 213–222, 2011, doi: 10.1016/j.swevo.2011.06.006.

[15] J. U. Heu, I. Qasim, and D. H. Lee, “FoDoSu: Multi-document summarization exploiting semantic analysis based on social Folksonomy,” Inf. Process. Manag., vol. 51, no. 1, pp. 212–225, 2015, doi: 10.1016/j.ipm.2014.06.003.

[16] J. P. Qiang, P. Chen, W. Ding, F. Xie, and X. Wu, “Multi-document summarization using closed patterns,” Knowledge-Based Syst., vol. 99, pp. 28–38, 2016, doi: 10.1016/j.knosys.2016.01.030.

[17] K. Wu, L. Li, J. Li, and T. Li, “Ontology-enriched multi-document summarization in disaster management using submodular function,” Inf. Sci. (Ny)., vol. 224, pp. 118–129, 2013, doi: 10.1016/j.ins.2012.10.019.

[18] S. Xiong and D. Ji, “Query-focused multi-document summarization using hypergraph-based ranking,” Inf. Process. Manag., vol. 52, no. 4, pp. 670–681, 2016, doi: 10.1016/j.ipm.2015.12.012.

[19] S. H. Zhong, Y. Liu, B. Li, and J. Long, “Query-oriented unsupervised multi-document summarization via deep learning model,” Expert Syst. Appl., vol. 42, no. 21, pp. 8146–8155, 2015, doi: 10.1016/j.eswa.2015.05.034.

[20] Y. Sankarasubramaniam, K. Ramanathan, and S. Ghosh, “Text summarization using Wikipedia,” Inf. Process. Manag., vol. 50, no. 3, pp. 443–461, 2014, doi: 10.1016/j.ipm.2014.02.001.

[21] E. Canhasi and I. Kononenko, “Weighted archetypal analysis of the multi-element graph for query-focused multi-document summarization,” Expert Syst. Appl., vol. 41, no. 2, pp. 535–543, 2014, doi: 10.1016/j.eswa.2013.07.079.

[22] D. R. Radev, H. Jing, M. Styś, and D. Tam, “Centroid-based summarization of multiple documents,” Inf. Process. Manag., vol. 40, no. 6, pp. 919–938, 2004, doi: 10.1016/j.ipm.2003.10.006.

[23] D. Wang, S. Zhu, T. Li, Y. Chi, and Y. Gong, “Integrating Document clustering and multidocument summarization,” ACM Trans. Knowl. Discov. Data, vol. 5, no. 3, 2011, doi: 10.1145/1993077.1993078.

[24] R. M. Aliguliyev, “Clustering techniques and discrete particle swarm optimization algorithm for multi-document summarization,” Comput. Intell., vol. 26, no. 4, pp. 420–448, 2010, doi: 10.1111/j.1467-8640.2010.00365.x.

[25] Y. Nie, D. Ji, L. Yang, Z. Niu, and T. He, “Multi-document summarization using a clustering-based hybrid strategy,” Lect. notes Comput. Sci., vol. 4182, p. 608, 2006.

[26] Y. Xia, Y. Zhang, and J. Yao, “Co-clustering sentences and terms for multi-document summarization,” Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 6609 LNCS, no. PART 2, pp. 339–352, 2011, doi: 10.1007/978-3-642-19437-5_28.

[27] X. Xu, “A New Sub-topics Clustering Method Based on Semi-supervised Learing.,” JCP, vol. 7, no. 10, pp. 2471–2478, 2012.

[28] Z. Yang, K. Cai, J. Tang, L. Zhang, Z. Su, and J. Li, “Social context summarization,” SIGIR’11 - Proc. 34th Int. ACM SIGIR Conf. Res. Dev. Inf. Retr., pp. 255–264, 2011, doi: 10.1145/2009916.2009954.

[29] J. Zhu et al., "Tag-oriented document summarization," WWW'09 - Proc. 18th Int. World Wide Web Conf., no. 2007, pp. 1195–1196, 2009, doi: 10.1145/1526709.1526925.
[30] S. Brin and L. Page, "Reprint of: The anatomy of a large-scale hypertextual web search engine," Comput. Networks, vol. 56, no. 18, pp. 3825–3833, 2012, doi: 10.1016/j.comnet.2012.10.007.
[31] G. Erkan and D. R. Radev, "LexRank: Graph-based lexical centrality as salience in text summarization," J. Artif. Intell. Res., vol. 22, pp. 457–479, 2004, doi: 10.1613/jair.1523.
[32] S. Hariharan, T. Ramkumar, and R. Srinivasan, "Enhanced graph based approach for multi document summarization," Int. Arab J. Inf. Technol., vol. 10, no. 4, pp. 334–341, 2013.
[33] S. Hariharan and R. Srinivasan, "Studies on Graph Based Approaches for Singleand Multi Document Summarizations," Int. J. Comput. Theory Eng., vol. 1, no. 5, pp. 519–526, 2009, doi: 10.7763/ijcte.2009.v1.84.
[34] X. Wan, "An exploration of document impact on graph-based multi-document summarization," EMNLP 2008 - 2008 Conf. Empir. Methods Nat. Lang. Process. Proc. Conf. A Meet. SIGDAT, a Spec. Interes. Gr. ACL, no. October, pp. 755–762, 2008, doi: 10.3115/1613715.1613811.
[35] F. Wei, W. Li, Q. Lu, and Y. He, "A document-sensitive graph model for multi-document summarization," Knowl. Inf. Syst., vol. 22, no. 2, pp. 245–259, 2010, doi: 10.1007/s10115-009-0194-2.
[36] Y. Gong and X. Liu, "Generic text summarization using relevance measure and latent semantic analysis," in Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, 2001, pp. 19–25.
[37] J. H. Lee, S. Park, C. M. Ahn, and D. Kim, "Automatic generic document summarization based on non-negative matrix factorization," Inf. Process. Manag., vol. 45, no. 1, pp. 20–34, 2009, doi: 10.1016/j.ipm.2008.06.002.
[38] D. Wang, T. Li, and C. Ding, "Weighted feature subset non-negative matrix factorization and its applications to document understanding," in Data Mining (ICDM), 2010 IEEE 10th International Conference on, 2010, pp. 541–550.
[39] E. Baralis, L. Cagliero, S. Jabeen, A. Fiori, and S. Shah, "Multi-document summarization based on the Yago ontology," Expert Syst. Appl., vol. 40, no. 17, pp. 6976–6984, 2013, doi: 10.1016/j.eswa.2013.06.047.
[40] L. Hennig, W. Umbrath, and R. Wetzker, "An ontology-based approach to text summarization," Proc. - 2008 IEEE/WIC/ACM Int. Conf. Web Intell. Intell. Agent Technol. - Work. WI-IAT Work. 2008, pp. 291–294, 2008, doi: 10.1109/WIIAT.2008.175.
[41] D. Wang and T. Li, "Weighted consensus multi-document summarization," Inf. Process. Manag., vol. 48, no. 3, pp. 513–523, 2012, doi: 10.1016/j.ipm.2011.07.003.
[42] J. Atkinson and R. Munoz, "Rhetorics-based multi-document summarization," Expert Syst. Appl., vol. 40, no. 11, pp. 4346–4352, 2013, doi: 10.1016/j.eswa.2013.01.017.
[43] G. Durrett, T. Berg-Kirkpatrick, and D. Klein, "Learning-based single-document summarization with compression and anaphoricity constraints," 54th Annu. Meet. Assoc. Comput. Linguist. ACL 2016 - Long Pap., vol. 4, pp. 1998–2008, 2016, doi: 10.18653/v1/p16-1188.
[44] D. Bollegala, N. Okazaki, and M. Ishizuka, "A bottom-up approach to sentence ordering for multi-document summarization," Inf. Process. Manag., vol. 46, no. 1, pp. 89–109, 2010, doi: 10.1016/j.ipm.2009.07.004.
[45] D. Bollegala, N. Okazaki, and M. Ishizuka, "A preference learning approach to sentence ordering for multi-document summarization," Inf. Sci. (Ny)., vol. 217, pp. 78–95, 2012, doi: 10.1016/j.ins.2012.06.015.
[46] J. Nasir, A. Karim, G. Tsatsaronis, and I. Varlamis, "A knowledge-based semantic kernel for text classification," in String Processing and Information Retrieval, 2011, pp. 261–266.
[47] D. Wang, S. Zhu, T. Li, and Y. Gong, "Multi-document summarization using sentence-based topic models," ACL-IJCNLP 2009 - Jt. Conf. 47th Annu. Meet. Assoc. Comput. Linguist. 4th Int. Jt. Conf. Nat. Lang. Process. AFNLP, Proc. Conf., no. August, pp. 297–300, 2009, doi: 10.3115/1667583.1667675.
[48] Y. Zhang, Y. Xia, Y. Liu, and W. Wang, "Clustering sentences with density peaks for multi-document summarization," NAACL HLT 2015 - 2015 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. Proc. Conf., pp. 1262–1267, 2015, doi: 10.3115/v1/n15-1136.
[49] B. Wang, J. Zhang, Y. Liu, and Y. Zou, "Density peaks clustering based integrate framework for multi-document summarization," CAAI Trans. Intell. Technol., vol. 2, no. 1, pp. 26–30, 2017, doi: 10.1016/j.trit.2016.12.005.
[50] N. K. Nagwani, "Summarizing large text collection using topic modeling and clustering based on MapReduce framework," J. Big Data, vol. 2, no. 1, pp. 1–18, 2015, doi: 10.1186/s40537-015-0020-5.
[51] J. Christensen, S. Soderland, G. Bansal, and Mausam, "Hierarchical summarization: Scaling up multi-document summarization," 52nd Annu. Meet. Assoc. Comput. Linguist. ACL 2014 - Proc. Conf., vol. 1, no. Figure 1, pp. 902–912, 2014, doi: 10.3115/v1/p14-1085.
[52] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," Science (80-.)., vol. 344, no. 6191, pp. 1492–1496, 2014, doi: 10.1126/science.1242072.
[53] D. Contractor, Y. Guo, and A. Korhonen, "Using argumentative zones for extractive summarization of scientific articles," 24th Int. Conf. Comput. Linguist. - Proc. COLING 2012 Tech. Pap., no. December, pp. 663–678, 2012.
[54] R. Ferreira, R. D. Lins, S. J. Simske, F. Freitas, and M. Riss, "Assessing sentence similarity through lexical, syntactic and semantic analysis," Comput. Speech Lang., vol. 39, pp. 1–28, 2016, doi: 10.1016/j.csl.2016.01.003.
[55] L. Marujo et al., "Exploring events and distributed representations of text in multi-document summarization," Knowledge-Based Syst., vol. 94, pp. 33–42, 2016, doi: 10.1016/j.knosys.2015.11.005.
[56] J. Carbonell and J. Goldstein, "Use of MMR, diversity-based reranking for reordering documents and producing summaries," SIGIR Forum (ACM Spec. Interes. Gr. Inf. Retrieval), pp. 335–336, 1998, doi: 10.1145/3130348.3130369.
[57] J. Lin, N. Madnani, and B. J. Dorr, "Putting the user in the loop: interactive maximal marginal relevance for query-focused summarization," in Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, 2010, pp. 305–308.
[58] M. G. Ozsoy, I. Cicekli, and F. N. Alpaslan, "Text summarization of turkish texts using latent semantic analysis," in Proceedings of the 23rd international conference on computational linguistics, 2010, pp. 869–876.
[59] N. Chatterjee and N. Yadav, "Fuzzy Rough Set-Based Sentence Similarity Measure and its Application to Text Summarization," IETE Tech. Rev. (Institution Electron. Telecommun. Eng. India), vol. 36, no. 5, pp. 517–525, 2019, doi: 10.1080/02564602.2018.1516521.
[60] J. Xu and G. Durrett, "Neural extractive text summarization with syntactic compression," EMNLP-IJCNLP 2019 - 2019 Conf. Empir. Methods Nat. Lang. Process. 9th Int. Jt. Conf. Nat. Lang. Process. Proc. Conf., pp. 3292–3303, 2020, doi: 10.18653/v1/d19-1324.
[61] C. Ding, X. He, and H. D. Simon, "On the equivalence of nonnegative matrix factorization and spectral clustering," Proc. 2005 SIAM Int. Conf. Data Mining, SDM 2005, no. 4, pp. 606–610, 2005, doi: 10.1137/1.9781611972757.70.
[62] E. Canhasi, "Query Focused Multi-document Summarization Based on Five-Layered Graph and Universal Paraphrastic Embeddings," in Computer Science On-line Conference, 2017, pp. 220–228.
[63] E. ShafieiBavani, M. Ebrahimi, R. Wong, and F. Chen, "On Improving Informativity and Grammaticality for Multi-Sentence Compression," no. November, 2016, [Online]. Available: http://arxiv.org/abs/1605.02150.
[64] E. Baralis, L. Cagliero, N. Mahoto, and A. Fiori, "GraphSum: Discovering correlations among multiple terms for graph-based summarization," Inf. Sci. (Ny)., vol. 249, pp. 96–109, 2013, doi: 10.1016/j.ins.2013.06.046.
[65] Y. Chali, S. A. Hasan, and S. R. Joty, "Improving graph-based random walks for complex question answering using syntactic, shallow semantic and extended string subsequence kernels," Inf.

Process. Manag., vol. 47, no. 6, pp. 843–855, 2011, doi: 10.1016/j.ipm.2010.10.002.
[66] A. John and M. Wilscy, "Vertex cover algorithm based multi-document summarization using information content of sentences," Procedia Comput. Sci., vol. 46, no. Icict 2014, pp. 285–291, 2015, doi: 10.1016/j.procs.2015.02.022.
[67] E. Canhasi and I. Kononenko, "Weighted hierarchical archetypal analysis for multi-document summarization," Comput. Speech Lang., vol. 37, pp. 24–46, 2016, doi: 10.1016/j.csl.2015.11.004.
[68] E. Tzouridis, J. Nasir, and U. Brefeld, "Learning to summarise related sentences," in Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, 2014, pp. 1636–1647.
[69] P. Sukumar and K. S. Gayathri, "Semantic based Sentence Ordering Approach for Multi-Document Summarization," Int. J. Recent Technol. Eng., no. 2, pp. 2277–3878, 2014.
[70] J. Christensen, Mausam, S. Soderland, and O. Etzioni, "Towards coherent multi-document summarization," NAACL HLT 2013 - 2013 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. Proc. Main Conf., no. June, pp. 1163–1173, 2013.
[71] R. Rautray and R. C. Balabantaray, "An evolutionary framework for multi document summarization using Cuckoo search approach: MDSCSA," Appl. Comput. Informatics, vol. 14, no. 2, pp. 134–144, 2018, doi: 10.1016/j.aci.2017.05.003.
[72] M. Das Anshuman Pattanaik, Santwana Sagnika and and B. S. P. Mishra, "Extractive Summary An Optimization Approach Using Bat Algorithm," vol. 904, pp. 339–351, 2019, doi: 10.1007/978-981-13-5934-7.
[73] P. Verma and H. Om, "MCRMR: Maximum coverage and relevancy with minimal redundancy based multi-document summarization," Expert Syst. Appl., vol. 120, pp. 43–56, 2019, doi: 10.1016/j.eswa.2018.11.022.
[74] A. Naserasadi, H. Khosravi, and F. Sadeghi, "Extractive multi-document summarization based on textual entailment and sentence compression via knapsack problem," Nat. Lang. Eng., vol. 25, no. 1, pp. 121–146, 2019, doi: 10.1017/S1351324918000414.
[75] A. Joshi, E. Fidalgo, E. Alegre, and L. Fernández-Robles, "SummCoder: An unsupervised framework for extractive text summarization based on deep auto-encoders," Expert Syst. Appl., vol. 129, pp. 200–215, 2019, doi: 10.1016/j.eswa.2019.03.045.
[76] I. M. and D. S. Abhishek Mahajani, Vinay Pandya, A Comprehensive Survey on Extractive and abstractive techniques for text summarization, vol. 904. Springer Singapore, 2019.
[77] N. B. and A. S. Vanyaa Gupta, Text Summarization for Big Data: A Comprehensive Survey. Springer Singapore, 2019.
[78] A. Kanapala, S. Pal, and R. Pamula, "Text summarization from legal documents: a survey," Artif. Intell. Rev., vol. 51, no. 3, pp. 371–402, 2019, doi: 10.1007/s10462-017-9566-2.
[79] A. Kumar and A. Sharma, "Systematic literature review of fuzzy logic based text summarization," Iran. J. Fuzzy Syst., vol. 16, no. 5, pp. 45–59, 2019, doi: 10.22111/ijfs.2019.4906.
[80] G. Erkan and D. R. Radev, "LexPageRank: Prestige in Multi-Document Text Summarization," in Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, 2004.
[81] J. Carbonell and J. Goldstein, "The use of MMR, diversity-based reranking for reordering documents and producing summaries," in Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, 1998, pp. 335–336.

Ms. Zakia Jalil is Assistant Professor in the Department of Computer Science and Software Engineering in International Islamic University, Islamabad, Pakistan. She received her master's degree in Computer Science in 2006. Currently, she is perusing her PhD degree in Computer Science, from International Islamic University, Islamabad, Pakistan.

Dr. Jamal Abdul Nasir is Assistant Professor in Department of Computer Science and Software Engineering in International Islamic University, Islamabad, Pakistan. He received his PhD from Computer Science Department of Syed Baber Ali School of Science and Engineering (LUMS), Lahore, Pakistan. His research directions include text categorization and clustering, text summarization, sentiment analysis, lexical expansions using contextualization, uplift modelling in Bioinformatics, data mining and machine learning algorithms and applications.

Mr. Muhammad Nasir is Lecturer Software Engineering in Department of Computer Science and Software Engineering in International Islamic University, Islamabad, Pakistan. He received his master's degree in software engineering in 2009 from Blekinge Institute of Technology, Karlskrona, Sweden. Currently, he is perusing his PhD degree in Computing, focused on Software Usability Engineering from Riphah International University, Islamabad, Pakistan. Previously, he worked in the software engineering industry in different positions, i.e., Research & System Analyst.