REVIEW OF RECENT TECHNIQUES FOR EXTRACTIVE TEXT SUMMARIZATION

[1]AHMED ELREFAIY, [2]AHMED RAFAT ABAS, [3]IBRAHIM ELHENAWY

[1]Teaching Assistant. Department of Computer Science, Faculty of Computers and Informatics, Zagazig University, 44519, Egypt

[2]Lecturer. Department of Computer Science, Faculty of Computers and Informatics, Zagazig University, 44519, Egypt

[3]Professor. Department of Computer Science, Faculty of Computers and Informatics, Zagazig University, 44519, Egypt

E-mail: [1]amnasser@zu.edu.eg

ABSTRACT

In the view of a significant increase in the burden of information over and over the limit by the amount of information available on the internet, there is a huge increase in the amount of information overloading and redundancy contained in each document. Extracting important information in a summarized format would help a number of users. It is therefore necessary to have proper and properly prepared summaries. Subsequently, many research papers are proposed continuously to develop new approaches to automatically summarize the text. “Automatic Text Summarization” is a process to create a shorter version of the original text (one or more documents) which conveys information present in the documents. In general, the summary of the text can be categorized into two types: Extractive-based and Abstractive-based. Abstractive-based methods are very complicated as they need to address a huge-scale natural language. Therefore, research communities are focusing on extractive summaries, attempting to achieve more consistent, non-recurring and meaningful summaries. This review provides an elaborative survey of extractive text summarization techniques. Specifically, it focuses on unsupervised techniques, providing recent efforts and advances on them and list their strengths and weaknesses points in a comparative tabular manner. In addition, this review highlights efforts made in the evaluation techniques of the summaries and finally deduces some possible future trends.

Keywords: *Extractive Text Summarization - Summarization Review - Artificial Intelligence - Information Retrieval - Natural Language Processing*

1. INTRODUCTION

Text summarization is the process of creating a shorter version of one or more documents that conveys the information in these documents. It produces a summary that reduces repetition in the text by containing a large part of the information in the original text. Therefore, we can say that summary is a tool that helps a user to efficiently find useful information from a vast amount of information [1]. The text started to be summarized in the late fifties [2] and yet, there has been considerable improvement in this area; and so, a large number of techniques have been proposed here [3], [4].

However, the generation of automatic text summarization is still a challenging task and more complex due to the issues founded in this task such as degree of redundancy, compression ratio which founded when summarizing multi documents than single document [5]. Furthermore, in recent years research seeks to overcome the lack of coherence presented by the summaries, resulting in common approaches identifying relevant content and integrating it into new parts of information [6], [7].

Another important aspect of summarizing the text relates to its Evaluation. There are several Methods have been presented to automatically evaluate summaries to link well with human evaluation. However, This is also a major challenge because it is not clear, even by human beings, what kind of information the summary should contain [8].

The two basic types of text summarization are abstractive and extractive [1]. Extractive summary extract the important and meaningful sentences from the original text and placing them into summary without any changes. Abstractive summary doesn't not rely on concatenating sentences; instead of that, it analyze the original text semantically to understand it and build more coherent meaningful related conclusion summary. The sentences in the summary may not be present in the original text. Abstractive summary give more generalized summary but it is difficult to compute.

Many researchers have presented comprehensive surveys about text summarization. Some of them still focusing on improving extractive text summarization and the others move toward abstractive summarization. Previously analysis on extractive text summarization presented elaborative studies for well-known approaches, recently discussed types or evaluation techniques to gain knowledge about text summarization key issues. In this paper, classification of extractive text summarization techniques is done into different new categories including supervised, semi-supervised and unsupervised. We focusing on unsupervised techniques, providing state of the art efforts and advances on them and list their strengths and weaknesses points in a comparative tabular manner. In addition, we highlight efforts made in the evaluation task and finally deduces some possible future trends. To our knowledge, we are the first to present such study for the unsupervised field in extractive text summarization.

In recent years, progress has been made in text summarization in various aspects, leading to the appearance of different subtypes under the two basic types. Based on the summarization Purpose, type of details or style of output, the summarization can be Indicative, Informative or Critical [9], [10]. Indicative summary present the main idea of the entire document, it gives the user a quick view from the original text. So, it may not contain all important factual content. Informative summary express the important concise information of the original text to the user. In Critical summary the document is criticized. For example, In the case of the scientific paper, it can expresses an opinion [10]. The most feasible type to automate is Indicative summary and the least one is Critical. Like Critical summary which can express an opinion for the document, scientific paper etc. Sentiment based summary generate summaries in the way that form opinion mining, according to person feelings towards the subject, product or entities. Many researchers have been presented comprehensive survey about this type of summarization [29], [30].

Based on the content type of the original text, the summarization may be considered as Generic or Query based [11], [12], [13], [9], and [14]. In Generic summary, Extracted information is not a user specific and doesn't rely on the document subject. In Query based summarization, the generated summary based on the user query. So, it present the user view. Query based summarization can be named as Topic-focused or user- focused summaries.

Based on the limitation of input text, summarization can be Genre specific, Domain Dependent or Domain Independent systems [9]. In Genre specific systems, specific inputs types only can be accepted such as, stories, newspaper articles etc. Domain Dependent systems deal with text which their subject defined in the fixed domain. Domain Independent systems can accept any type of text as they are not relied on the domain.

Based on the number of input documents, in which the system input can be one or more documents [9]. It can be divided into Single Document or Multi-Document Summarization [7], [15]. In Single Document Summarization, summarization is built on one document only whereas in Multi-Document Summarization, summarization is built based on more than one document, all of them are of the same topic. Multi-document summarization may suffer from some issues such as redundancy, sentence ordering, temporal dimension, co-reference which make this task more difficult than summarizing task of single document type. [5].The most prominent issue which also appeared more with multi-document task is redundancy. So, there are some attempts to tackle this problem such as selecting the sentences at the beginning of the paragraph and then measure the similarity of the following sentence with the sentences already chosen and this sentence is retained only if it consists of new related content [16]. Maximal Marginal Relevance approach produced at 1998 [17]. Another different methods suggested by researchers trying to achieve best results in multi-document summarization [18], [13], [19], [20], [21], and [22].

Based on the language of the text, which the system can accept. Summarization can be Mono Lingual System, Multi Lingual or cross-lingual

System. Mono Lingual System deal with documents with specific language and the produced summary is based on that language. In Multi-Lingual System, source documents are more than one language and generated summary are in these different languages. In cross-lingual, the input document is in specific language and the output is in a different language than input language.

Based on the level of linguistic space. Summarization approaches can be either Shallow Approach or Deeper Approach [23]. Shallow approaches limited on syntactically representation and try to extract the prominent parts of the text. Deeper approach restricted on semantically representation and basically depend on linguistic processes during the extraction method.

Every year the Web pages increase significantly and there are some of search engines return list of web pages as a result for a single search query. Users usually need to know which documents are relevant and which are not through going through multiple pages. In addition, they are abandoning the search in the first attempt. Therefore, it's important to generate summaries and pick up important information in web pages. Such summaries are web-based summaries. WebInEssence is a search engine which can generate summaries from clusters of related documents [24]. Due to e-mail overloading problem that happens when e-mails keep coming in the inbox and great time consuming in reading or archiving them, there is a need to summarize email conversations. Such type of summarization is called E-mail based summarization.

Summarization also can be Personalized which generate summary of information related to the user interests. Therefore, the summary system need to keep tracking with user profile to be able to determine relevant information that the user is interested in. User profile can be determined by statistical mapping method from personality characteristics such as genders with some other features [25]. Another different methods suggested by researchers using this type of summarization [26], [27]. Update based summary generate summaries by acquiring the latest updates related to the topic by taking into considerations that users already have fundamental knowledge on the subject [28]. Survey summaries are another kind which present a long overview for a specific subject or entity, trying to gathering the most significant facts belonging to any entity, person, place etc. Survey summaries contain these types: Wikipedia articles, Survey summary and biographical summary [31].

2. EXTRACTIVE TEXT SUMMARIZATION BACKGROUND

Extractive text summarization done by picking up the most important sentences from the original text in the way that forms the final summary. Extractive techniques generally generate summaries through 3 phases or it essentially based on them. These phases are *preprocessing step*, *processing step* and *generation step*:

1) Preprocessing step: the representation space dimensionality of the original text is reduced to involve a new structure representation. It usually includes:

a. Stop-word elimination: Common words without semantics that do not collect information relevant to the task (for example, "the", "a", "an", "in") are eliminated.

b. Steaming: Acquire the stem of each word by bringing the word to its base form.

c. Part of speech tagging: The process of identifying and classifying words of the text on the basis of part of speech category they belong (nouns, verbs, adverbs, adjectives).

Another technique used here is case folding, in which all characters are converted to the same kind of letter case, either lower case or upper case [23] . But, it's not good to use this technique when dealing with documents in domains which suppose for example that the appearance of upper case word in the sentence increase its importance [32]. Finally in this phase, the sentences are analyzed and transformed in terms of features to be ready for the next stage. The sentences are analyzed on the basis of statistical, linguistic or hybrid analysis of features where statistical features doesn't take into consideration word meanings but, linguistic features goes deeply to capture semantic meanings. Each sentence in the document is transformed in terms of these features so that we can determine whether it is important enough to include it in the summary or not. Table 1 below shows extractive text summarization common features and table 2 below shows comparison between extractive text summarization statistical and linguistic features.

2) Processing step: It uses an algorithm with the help of features generated in the preprocessing step to convert the text structure to the summary structure. In which, the sentences are scored.

3) Generation step: sentences are ranked. Then, it pick up the most important sentences from

Table 1: Extractive Text Summarization Common Features

Features	Description	Comments
Sentence Position	It implies that in a specific position, the important sentences will be presented such as first or last positions.	Ex. Value = 1 for first or last position. Otherwise, equation can be used to keep tracking with remaining positions to take values between 0 and 1 [36], [37]
Title Similarity	The sentence is considered to be important if it has similarity with the document title. This similarity can be calculated by cosine similarity measure.	Can't use this feature with documents without title
Similarity to Keywords	Compute the similarity between each sentence and set of keywords based on the cosine similarity measure.	Can be used with query-based summarization
Sentence Length	Sentences with specific length are considered to be important.	Generally shorter and longer sentences have small values as they are not suitable for the summary [38].
Term Frequency	This means terms that have occurred over and over and that increase the score of their sentences. It reflects how important the word is for the document.	The term word in term frequency feature can take several views such as unique term or word, Bi gram key or tri gram key [37]. It can be calculated by the number of occurrence for the term. The most term frequency measures used are TF-IDF [39], [38] and TF-ISF [25], [37]. TF-IDF or TF-ISF means that the terms in unit (e.g. document or sentence) are important only if they are not appeared more frequently in the whole collection of all units.
Cue Method	Words that have positive or negative effect on sentence weight.	Such as: in conclusion, in summary [38].
Proper Noun	Sentences which have proper nouns are considered to be important.	Such as: name of a persons, organizations or places [37], [38].
Sentence to Sentence Similarity	The similarity between each sentence and all other sentences calculated, added up and then normalized [37].	This feature employs the concept of text coherence [25].
Sentence to Centroid Similarity	Centroid sentence is calculated first. Then similarity between each sentence and the centroid sentence calculated [37].	This feature employs the concept of text coherence [25]. For example, centroid sentence is calculated on the basis of TF-ISF feature – sentence with highest TF-ISF value is considered to be centroid [37].
Numerical Data	The Appearance of such data in a sentence can reflect important statistics and can increase its chance to be selected for the summary.	[32], [37]
Presence of Special Characters or Words	Some of them give the sentences lower probability to be selected such as: presence of brackets. And others give the sentences higher probability such as: presence of commas, inverted commas, acronym words and upper case words.	[32], [40]

Table 2: Comparison between Extractive Text Summarization Statistical and Linguistic Features

Type	Statistical features	Linguistic features
Description	Doesn't take into consideration word meanings; instead of that, it try to analyze and extract sentences using statistical features only.	It goes deeply to be aware of the semantics connections between words and know the linguistic knowledge. It identifies term relationships through part of speech tagging, grammar analysis and other techniques.
Examples	Term frequency, sentence length and position, cue method, title method, etc.	Lexical chain, word net, Transition relationship, Anaphoric relationship, etc.
Advantages	Efficient in computations.	Based on the semantic meanings. Generate better summary results.
Disadvantages	Lack of the semantic meanings.	Computations take more time than statistical. It is difficult to compute rather than statistical.

the ranked structure to generate the final required summary.

The last two stages - processing and generation steps - can be also described approximately as three main components: sentence scoring, selection and paraphrasing (reformulation).

At sentence scoring, for each sentence a score is assigned which points to its significance. After that, the most important sentences is extracted. Sentence scoring can be done via several approaches: supervised, semi-supervised or unsupervised approaches (cf. Sect. 4). At sentence selection, the summarization system has to specify the best collection of significant sentences that form the final summary with taking into consideration the most prominent factors: redundancy and cohesion. The traditional method for sentence selection is to pick up the top ranked sentences directly but, the redundancy elimination is the key issue especially for multi-document summarization. There are more than one approach used for this task (sentence selection). For instance, Maximum Marginal Relevance (MMR) is the most popular approach for such task [17] which find the linear incorporation for relevance and novelty –independently– measures. Another approaches based on the Kullback–Leibler (KL) divergence in which sentences are selected in the way that decrease the KL divergence between words probability distribution of the candidate summary and probability distribution from the input [38], [39]. And because decreasing KL divergence are mathematically tenacious, it is optimized via greedy selection.

At sentence paraphrasing (reformulation), the selected sentences to form the summary are modified or reformulated in order to enhance the summary, provide more cohesion and clarity and also eliminate redundant or unnecessary information, for example the usage of reformulation and sentence fusion [6].

The summarization process main phases can be discussed by another view in which it contains the following three main subtasks: topic identification, interpretation and finally the summary generation [40].

3. TEXT SUMMARIZATION EVALUATION

Performance measurement (evaluation) of the automatic summaries is a challenging task. Due to manual evaluation is difficult and time consuming, a lot of techniques have been made to automate evaluation task. Evaluation can be computed by two ways:

1. Extrinsic evaluation: evaluation of summary done based on how it provides help to other tasks. It includes several methods like:

a. Relevance assessment: it evaluate the relevance of a topic in the summary or original text.

b. Reading comprehension: it represents the capability or correctness of answering multiple choices questions that can be gathered after reading summary.

2.Intrinsic evaluation: it depends on human judgment as, it evaluate the summary based on the coverage of this summary (system summary) and the human-written summary and so, the evaluation of the summary can be Quality or informativeness.

a. Informativeness evaluation: it is computed by comparing system summary with human-written summary or comparing the system summary with the original text to check that the summary contains similar contents as original text. It includes: ROUGE [41], [42], Relative utility [43], Factoid Score [44], Pyramid Method [45], etc.

b. Quality evaluation: it is provided based on linguistics so expert humans evaluate summaries manually based on five linguistic questions including: non redundancy, focus, grammaticality, referential clarity, and structure and Coherence. Due to none of the previous questions can be properly modeled automatically; thus, manual evaluation is irreplaceable.

Recall-Oriented Understudy for Gisty Evaluation (ROUGE) [41], [42] is the standard method to evaluate summarization automatically. It is based on the comparison of n-grams between the system summary (to be evaluated) and reference summaries (human-written summaries). ROUGE metrics have more than one shape including: ROUGE-N (refer to n-grams), ROUGE-S (skip bigrams), ROUGE-L (longest common subsequence), ROUGE-W (weighted longest common subsequence), or ROUGE-SU (skip bigrams and unigrams). The most commonly used one is ROUGE-N, in which n-gram based metrics are computed with the recall, precision and f-measure oriented score as following:

$$ROUGE - N_{recall} = \frac{\sum_{s\in ref\_sum}\sum_{Ngram\in s} Count_{match}(Ngrams)}{\sum_{s\in ref\_sum}\sum_{Ngram\in s} Count(Ngrams)} \quad (1)$$

$$ROUGE - N_{precision} = \frac{\sum_{s\in ref\_sum}\sum_{Ngrams\in s} Count_{match}(Ngrams)}{\sum_{s\in cand\_sum}\sum_{Ngram\in s} Count(Ngrams)} \quad (2)$$

$$ROUGE - N_{F-score} = \frac{2\,x\,ROUGE - N_{recall}\,x\,ROUGE - N_{precision}}{ROUGE - N_{recall} + ROUGE - N_{precision}} \quad (3)$$

4. EXTRACTIVE TEXT SUMMARIZATION TECHNIQUES

From the late fifties until now, there are several extractive text summarization techniques which can be classified based on its nature into 5 approaches: Statistical, Graph, Machine-learning, Fuzzy-Logic and Latent Semantics approach and additionally into topic, discourse approach which come from or based on one or more from the previous approaches. These approaches can be categorized based on learning type into supervised, semi-supervised and unsupervised approach. Figure 1 below show the extractive text summarization techniques categorized by learning type.

4.1 Statistical Approaches

Earlier approaches mostly depend on statistical approaches, mainly on frequency and centrality, also frequency and centrality are earlier unsupervised approaches. The assumption is that the most significant information will contain the most frequent words. Luhn [2] generated summaries based on term frequency to detect the importance of a sentence in the document. There are many techniques based on term frequency feature include another statistical features with it. For example in [46], single document summarization generated based on the combination of word-frequency feature (WF), Textual Entailment (TE), and The Code Quantity Principle (CQP). Hence, there are many established features that can be used with statistical approaches - such as: sentence position, positive and negative terms, title similarity, sentence centrality, term frequency, etc - which can be used to score the sentences and then pick up the highly scored ones to generate the final summary. Another features which can detect word or term importance are: TF*IDF (Term Frequency-Inverse Document Frequency) [47], information-gain [48] which used to detect the relevance of terms or sentences, mutual-information which used to measure dependency or information shared between two terms and residual-IDF (residual-inverse document frequency) in which term frequency is calculated based on Poisson distribution. In [49], summarization generated based on some features including similarity to centroid sentence in which centroid sentence captured based on TF-IDF and then each sentence calculated the similarity value with the centroid based on cosine similarity and then features values for each sentences added together to get sentences scores. A detailed review of techniques based on statistical approaches are discussed in [50], [51].

4.2 Graph Approaches

In statistical approaches, central sentences that have maximum similarity to others, supposed to contain the most central-ideas of the text. The previous assumption helps to form the foundation of graph based approach. Graph based ranking approach is based on Page-Ranking algorithm [52] in which text unites (words or sentences) are represented by nodes in a weighted graph, with weighted edges determined using similarities between nodes. Both TextRank and LexRank are graph based approaches. In TextRank [53], importance scores of nodes determined used voting based weighting while in LexRank [54], it's a cosine-transform-based weighting algorithm. TextRank was introduced as the first graph based approach algorithm in which a vertex obtains more significance if it connects with a higher number of vertices as each vertex casts voting to the connected vertex with it. Mihalcea introduced TextRank for sentences extraction and keywords extraction of single document task, while LexRank is for multiple-documents task. A graph is formed for all sentences as nodes, and for each two nodes if they are similar to each other's with a value greater than a threshold then they can be connected. After graph is made, a random walk is occurred to detect highly central sentences. In 2007 [55], an approach that relied on affinity-graph was introduced for generic-based and topic-based multi document summarization. Summarization done by picking up the highest information richness and novelty by calculating similarities on differentiating intra document and inter document connections between links. After that greedy algorithm used to penalize redundancy.

In the last few years, there are several researches proposed based on the graph approaches which also presented good results in summarization. For instance, GRAPHSUM [56] was developed in 2013 as graph based summarizer for novelty and general purpose in which association rules is performed to discover correlations between terms. Recent graph based approaches relies on lexical association for determining document topic. Murali in 2016 [57] proposed technique based on lexical association with the help of graph-based ranking algorithms to assign relative weights for the retrieved keywords which used after that in sentences scoring. Ravinuthala in 2016 [58] assumed that the topics are formed by identified words and then the central idea formed through the topics, called theme. So, the technique depends on lexical association relationship to extract words that form document themes. TextRank and LexRank are fully-unsupervised algorithms as they didn't rely on training set but rather they depends on the entire text.

4.3 Machine-Learning Approaches

Variety of techniques based on machine-learning approaches are proposed which can be classified into supervised, semi-supervised or unsupervised approach. Supervised approaches needs training datasets (labeled data) represented in a set of documents with their human summaries so, it can be easily to learn and detect important features of the sentences. Supervised learning techniques are such as Regression, Multilayer Neural network, Decision Tree, Support Vector machine, Genetic Algorithm and Naïve Bayesian Classier. Semi-supervised approaches depends on labeled and unlabeled data to produce the convenient classifier; For instance, Support Vector machine (SVM) and Naïve Bayes Classier are used as semi-supervised learning techniques [59]. On the other hand, unsupervised approaches generate summaries without needing of training data. Hidden Markov Model, Clustering and Deep learning techniques (RBM, Autoencoder, Convolutional network, RNN) are instances of unsupervised learning technique.

The earlier machine-learning techniques used are binary classifier, Bayesian method [60] and Hidden Markov Model. In Binary Classifier using Bayes' rule [61], the probability to include the sentence in summary is calculated for each sentence given some features. And for Hidden Markov Model [62], the algorithm detects a likelihood of each sentence to be included in the summary. Also in 2002 [63] a summarization algorithm proposed based on Logistic Regression Model (LRM) and Hidden Markov Model (HMM) using a joint distribution to the features collection rather than the assumption of features independency in Naive Bayesian techniques. And for this assumption, HMM have advantage over Naive Bayesian algorithm.

In 2005 [64], RankNet was discussed, a gradient descent method using Neural Network to learn the ranking function that used in sentences scoring. Based on RankNet, NetSum was developed on 2007 [65], two layer neural network trained by RankNet - actually RankNet here was implemented in a more enhanced algorithm called LambdaRank - to score sentences and then pick up the highest ones. LambdaRank framework [66] is a flexible enhanced algorithm for ranking which works through non smooth target cost function, providing a training speed up and more accuracy. Support Vector Regression (SVR) algorithm is used in [67], based on some features (such as sentence position, name entities, semantic features, word and phrase features) in which the model trained to score text sentences. Support Vector Machine (SVM) was used in [68] for query-based summarization to reveal the relevant sentences to be inserted in the final summary. Also in 2009 [69], structural SVM used to summarize a single document taking into consideration diversity, coverage, and the balance issues. A trainable summarizer was proposed in 2009 [15], focused on some features including sentence position, sentence centrality, positive and negative word, Bushy path of node (sentence), etc. And the following models including: GA, Mathematical Regression, Feed Forward NN, Probabilistic NN and Gaussian Mixture Model are used to train previous features. Also Fattah and Ren discuss the effects of each feature and showed that the sentence Bushy path feature is the most significant one, also showed that Gaussian Mixture Model results outperform other models results. In 2014 [70], multi-document summarization technique based on hybrid model of Maximum Entropy, Naïve Bayes and SVM which are trained on some features to score sentences and then form the final summary. Another algorithm to summarize single mono-lingual documents based on Memetic Algorithm (MA) is [71], in which genetic operators is used with the help of local search strategy, called MA-SingleDocSum and this technique outperformed state of the art methods. Another technique for summarization belonging to supervised approaches is Conditional Random Field (CRF), a popular probabilistic model that focusing on machine-learning and used for structured prediction. CRF in [72], used as a sequence labelling problem to detect the correct features that include the interactions between sentences.

On the other hand, a great efforts in unsupervised machine-learning approaches occurred on the last years and updated continuously. Starting with HMM as we mentioned before [62] where HMM detects the probability that each sentence should be included in the summary. Based on some statistical features including sentence position, number of terms, baseline term probability and document term probability, calculate the posterior probability that each sentence can be picked up to be in the summary. The algorithm handle naïve Bayes classifier limitations by some dependency assumptions, including sentence positional dependency, dependency among all features and dependency between each two sentences where the probability to select one sentence to be in the summary depends on the status of the previous one (it was included or not in the summary), called Markovity.

In [73], Fung and Ngai proposed a new unsupervised training multi document summarization technique which can be used to generate summaries by picking up the prominent sentences or used to detect topics. The proposed method combines vector space clustering model via modified K-means for iteratively classifying articles and segmental K-means decoding for paragraph and sentences classifications and tagging data into sentence-class pairs with a probabilistic model via Hidden Markov Model for sentences cohesion and clustering improvements. And then, it's easy now to extract the prominent sentences from each theme (class) for the final summary.

In recent years, leap occurred in unsupervised machine learning approaches; especially in clustering, deep learning techniques. A query-based document summarizer based on OpenNLP tool and Clustering technique is presented in [74]. The summarizer obtain paragraphs from the document and build document graph, where nodes represent paragraphs and edges represent syntactic relationships between nodes which calculated by semantic parsing. After that, K-mean clustering algorithm applied to group coherent sentences with each other based on associativity degree according to keywords in the user's query. Finally, picking up the top five nodes to form the final summary.

And in [32], another clustering based approach technique discussed to summarize query-based multi documents, in which the documents are clustered using cosine similarity; then sentences within each document-cluster are clustered and then pick up the best sentences from each sentence-cluster. This paper introduced the user query strengthening where the most repeatedly words in documents are picked up and added to the query. Furthermore, the cosine similarity between each sentence and the query are calculated to accurately select best sentences for the summary.

Despite, researchers face difficulty to cluster sentences compared to clustering the documents. Louvain clustering algorithm was introduced with the help of dependency graph for single document summarization [75]. The algorithm build dependency graph for sentences and applying Louvain algorithm for words clustering so, words within each cluster are scored based on the dependency relations. Furthermore, scores of words are strengthened and enhanced by several approaches, including increasing word score by one if it was mentioned in the context of another keyword (related keyword), and also adding term frequency score of each word to its scores. After that, sentence score is calculated by the summation of its words scores, and then top sentences in scores are selected to form the summary.

Another single document summarization approach based on Agglomerative clustering is proposed in [76]. After the document is preprocessed, it is represented by Vector Space Modeling and the weights are assigned using TF-ISF measure. After that, Agglomerative nested clustering (hierarchical approach) applied for sentences clustering based on cosine similarity measures and then sentences within each cluster are scored based on sentence similarities with other sentences in its cluster added to sentence similarity with the title. Finally, from each sentence-cluster, pick up top two ranked sentences for the final summary. (Disadvantage here: lack of coherence).

Moreover, Deep Learning Techniques represented by Boltzmann machines [77], [78], [34], Auto-Encoder [79], Convolutional Neural Network [80], [81], [82] and Recurrent Neural Network [83] are recently proposed in summarization field. The first paper that uses Deep Learning technique is [77], in which a Deep Boltzmann machine is utilized for query oriented multi document summarization. This algorithm tries to predict concept importance via Query Oriented Deep Extraction (QODE); a three stages of Deep Belief Network (DBN): concept extraction, reconstruction validation, and summary generation. In first stage, DBN is used to filter out not important words and discover others through DBN layers. Then, apply fine tuning process (for reconstructing distribution of data) to get important sentences. And finally, Dynamic Programming (DP) is used to maximize summary importance that make summary length equal to 250 words.

In [78], Restricted Boltzmann machine (RBM) is used with two hidden layers where each sentence represented by four features including title similarity, sentence position, term weight and concept feature and so RBM input is sentences features vector. RBM aim to refine sentences by get optimal feature vector set and then score sentences by calculating intersection between each one and user query, after that ranking sentences and select top sentences for the summary. Depending on the previous algorithm, another technique for single document summarization proposed [34] where features increased to be eleven-feature vector values including sentence position, TF-ISF, sentence to sentence and centroid similarity, named entity, etc.

In [79], a Deep Auto-Encoder technique is used for extractive query-based single document summarization and based on local term frequency feature the AE tries to detect and learn the features and then rank sentences using cosine measure with subjects or key phrases. Unlike others deep learning techniques which may suffer from sparse input representation, this technique proposed solutions to reduce this problem via two techniques. First, developing local word representation (a bag-of-words (BOW) representation) consisting of input representations of each sentence in the document and second, additional random noise value added to the word representation weight. Also in this paper, another a Deep Auto-Encoder technique based on ensemble approach called Ensemble Noisy Auto-Encoder (ENAE) is used in which the model runs multiple times on the same input, each with different added random noise to input representation. This led to different extractive summaries and then aggregate the ranking of these different experiments, after that sentences that occur most frequently are obtained to form the final summary.

In [80], Convolutional neural network (CNN) is applied for multi document summarization to model and project sentences into distributed representation and then cosine similarity measurement is applied for representing and modeling the sentences redundancy. After that, sentence selection method called diversified selection is used as an optimization problem to pick up the high quality sentences by minimizing prestige and diversity cost of them. PriorSum model is proposed in [81] to determine the chance of the sentence to be selected in a summary without considering its context. An enhanced CNN is applied to learn the overall set of document independent features from variable-length phrases. The enhanced CNN applies two max-over-time pooling operations, first one to detect the most prominent features and the second to capture the best representative features. After that, the generated independent features are combined with document dependent features such as position, term frequency and cluster frequency and working after that with the regression model [67] for ranking sentences. A query focused multi-document summarization model based on CNN is discussed in [82], where the model use weighted-sum pooling over sentence embeddings to represent document cluster by learning query relevance of the sentence (from attention over sentence representations based on the query). After that, sentences are ranked using their similarity representation to the document cluster.

In [83], a Recurrent Neural Network (RNN) based on Gated Recurrent Unit neural network (GRU) is proposed to handle single document extractive summarization as sequence classification task in which a binary decision is computed for each sentence (taking into consideration the previous decision made) to detect whether it should be selected or not.

4.4 Latent Semantic Analysis Approaches

Latent Semantic Analysis (LSA) is considered a fully-unsupervised method for learning and representing the contextual usage meaning of words by statistical computations; so, it has the ability to avoid the problem of synonymy by using semantic content of words. LSA composed of three main steps including: input matrix creation, singular value decomposition (SVD) and sentence selection. In input matrix creation, the input document is represented by a matrix in which columns are mapped to sentences, rows are mapped to words and cells represent importance of words in sentences. The function that calculate cells values is called a weight function which can be Normal, GFIDF, IDF or Entropy weight function [84]. In singular value decomposition, to model the relationship between words and sentences as it decompose the input matrix into three other matrices (first and third matrices represents vector of extracted values for the original rows and original columns respectively and the second matrix represents scaling values and the third matrix represents original columns as vector of extracted values). In sentence selection, important sentences are selected from SVD results, different algorithms used here like Gong and Liu [11], Steinberger and Jezek [85], Murray, Renals and Carletta [86] and Ozsoy [87]. A comprehensive survey about these algorithms have been presented here [88]. Table 3 below shows these methods in comparative manner.

Latent Semantic Analysis-based Text relationship map (LSA + TRM) is proposed for automatic summarization [89], in which LSA is used to obtain text’s semantic matrix and build relationship map based on sentence’s semantic representation. After that, a global bushy path is used to select important sentences to generate final summary. A multi-document Summarization technique was proposed based on Optimal Combinatorial Covering Algorithm (OCCAMS) [90] and outperforms all human generated summaries (CLASSY11). OCCAMS is based on LSA algorithm to learn terms distribution for documents and then use optimization methods (greedy methods for Budgeted Maximal Coverage and dynamic programming method Fully Polynomial Time Approximation Scheme) for maximizing combination of covered terms weight and minimizing redundancy.

Table 3: LSA sentence selection algorithms

LSA algorithm	Main Steps	Selection Criteria
Gong and Liu's Method [13]	1. Input matrix creation. 2. SVD Calculations. 3. Sentence Selection.	Based on matrix V^T.
Steinberger and Iezek's Method [85]		Based on matrix V^T and length of sentence vector.
Murray, Renals and Carletta's Method [86]		Based on matrix V^T and ∑ matrices.
Ozsoy's (Cross Method) [87]	1. Input matrix creation. 2. Preprocessing.	Based on matrix VT, average value of each sentence and length of each sentence.
Ozsoy's (Topic Method) [87]	3. SVD Calculations. 4. Sentence Selection.	Based on matrix VT, creation of concept x concept matrix, strength value of each concept and discovering the main and sub concepts.

4.5 Fuzzy-Logic Approaches

Some of the features used in the previous summarization approaches such as main concepts, occurrence of anaphors and proper nouns have binary values (zeros and ones) which sometimes are not exact. To solve this problem, these binary feature can be redefined as fuzzy quantities to take values ranging from zero to one [91]. Fuzzy logic are able to model common sense reasoning in addition to dealing with uncertainty in an unsupervised manner. On the other hand, the classification solution is another task appeared using fuzzy logic to summarize text. For instance, in [92], fuzzy-rough set aided method is proposed to extract key sentences, in which approach the sentences takes relevance ranking based on fuzzy relevance clustering. The relevance of each sentence is maintained by a vector of these features: sentence position, length, TF-ISF and semantic pattern, after that these vectors are clustered by fuzzy c-mean algorithm (FCM) and the relevance score is computed for each sentences. Finally, pick up sentences with relevance score larger than 0.5 to be candidate sentences and then select highest scored sentence from each cluster to form the final summary. This method tackle the problem of “sentences of similar semantic meaning but written in synonyms are treated differently” by depending on senses rather than raw words.

In [93], a single document summarization approach is discussed based on nine features including sentence centrality, position, length, number of proper noun, etc with using the combination of fuzzy rules and sets to pick up sentences based on their features. On the other hand, there are some researches supposing that integration of fuzzy logic with other approaches will give better results, such as previously mentioned approach which integrate fuzzy set with rough set [92]. Another integration approach was proposed in [94], which incorporated fuzzy logic with swarm intelligence where features weights is obtained from the swarm algorithm to adjust features score and use them as inputs for the fuzzy

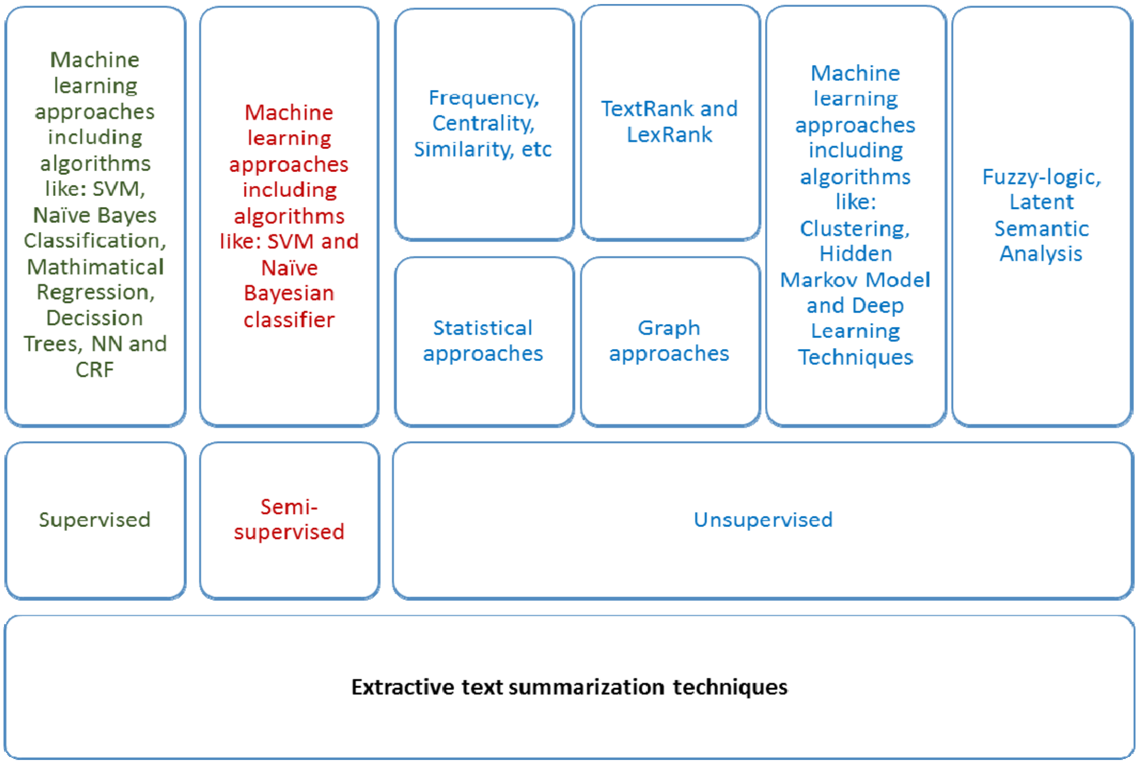
Machine learning approaches including algorithms like: SVM, Naïve Bayes Classification, Mathimatical Regression, Decission Trees, NN and CRF
Machine learning approaches including algorithms like: SVM and Naïve Bayesian classifier
Frequency, Centrality, Similarity, etc
TextRank and LexRank
Machine learning approaches including algorithms like: Clustering, Hidden Markov Model and Deep Learning Techniques
Fuzzy-logic, Latent Semantic Analysis
Statistical approaches
Graph approaches
Supervised
Semi-supervised
Unsupervised
Extractive text summarization techniques

Figure1: Taxonomy of Extractive Text Summarization Techniques Categorized by Learning Type

Table 4: Advantages and Disadvantages of Extractive Text Summarization Approaches

Techniques	Advantages	Disadvantages
Statistical based approaches	1. Simple and fast processing. 2. Requires less processor and memory capacity. 3. Unsupervised approaches, no need for training datasets.	No linguistic knowledge processing or semantic relation mapping [98].
Graph based approaches	1. Can generate query-specific or topic-specific summaries. 2. Unsupervised approaches, no need for training datasets.	Accuracy will rely on the selected affinity function.
Machine-learning based approaches	1. Simple. 2. Easy to test performance of high number of features.	1. Requires statistical data. 2. Need a huge training corpus for supervised and semi- supervised techniques.
Latent Semantic Analysis approaches	1. Provide Semantic relation. 2. Present important information with least noise.	Difficult to handle polysemy.
Fuzzy logic based approaches	1. Knowledge-driven reasoning based, can take better results if integrated with data-driven technique. 2. Fuzzy logic can give compression ratio as low as 20%.	1. Human experts are needed to define the fuzzy rules. 2. Overhead in designing the membership function.

inference system to gather the final scores. In [95], fuzzy logic approach integrated with latent semantic analysis (to keep aware of text semantics) for single document summarization where each approach generate a summary and then intersect both summaries to find the final one. Like the previous technique, another one is proposed in [96] where fuzzy logic, bushy path and WordNet synonyms are used, each algorithm give different summary and then find the intersection of these summaries to form the final summary.

In [97], Adaptive Neuro-Fuzzy Inference System (ANFIS) – that is used to summarize single documents – is a fuzzy inference system which implemented based on the frameworks of NN. A vector of nine features for each sentence including: title similarity, sentence position and similarity, numerical data, proper noun, etc will be input to nine neurons in ANFIS model. After that, each input converted to a fuzzy value using membership function which then used to compute the firing strength of the corresponding rule. ANFIS model contained premise and consequent parameters for the IF and THEN that will be adjusted during the training based on a combination of least-square estimation and back-propagation gradient descent method. The ANFIS model is learned to be able from classifying sentences as summary and non-summary sentence. This model tackle the problem of needing the human experts for building fuzzy rules by using subtractive clustering method to automatically generate rules.

5. COMPARING UNSUPERVISED EXTRACTIVE TEXT SUMMARIZATION TECHNIQUES

While, there are many approaches for extractive text summarization, each approach still suffer from some limitations.

Statistical approaches have simple and fast processing without the need for training datasets, but they generate summaries with no linguistic or semantic knowledge. Graph approaches can generate query or topic specific summaries with good information coverage, but the accuracy depends on the used affinity function. Machine-learning approaches can represent document features in appropriate manner, test the performance of high number of features, providing a solution for sentence scoring problem, but it is recommended to use statistical data and a huge training datasets to generate high accuracy summaries. Latent semantic approaches are the best to provide semantic relations and generate a good coverage knowledge with least noise, but they still suffer from polysemy problem. Fuzzy logic approaches are good alternative to improve sentence scoring problem and enhance summarization if integrating with other techniques, but human experts are need to define fuzzy rules.

Therefore, to handle the limitations of given approach, it can be integrated with another helper technique to improve the accuracy of the summary. For instance, the usage of Fuzzy c-mean clustering technique in [92] which reduce the redundancy and give good information coverage. Integrating of Fuzzy-Logic with LSA [95] which handle sentence scoring problem and LSA that generate semantic summaries. Also Greedy algorithms or Dynamic programming techniques can be integrated to handle sentence selection task to achieve high coverage and low redundancy [55], [56], [90].

Table 4 above shows advantages and disadvantages of the previous discussed 5 approaches.

The recent unsupervised techniques that have been discussed above (under the 5 approaches in Sect. 4) are compared in a tabular form with additional details about them. Table 5 below shows such a comparison of these unsupervised extractive text summarization techniques.

In text summarization supervised training approaches, there is a need to obtain human labeled class-sentence pairs to complete training and testing operations; but, hand labeling large collection of documents with theme-classes is very tedious and time consuming task. In addition, there is a huge amount of dispute between humans on manual labeling (annotation) of document themes and topics. How many themes or topics should be present? What's the beginning and ending of each topic? Therefore, it would be better to learn and decode the hidden theme or topic of text using an unsupervised training method without manually labeled data (manually annotated data) and this is the first reason why we focus on unsupervised approaches.

The second reason is that in supervised training approaches and given any corpus datasets, it's possible to learn corpus rules and features by training and testing; but, such approaches become corpus-based approaches which cannot guarantee that the generated summaries are helpful, due to its shortage of coherence and cohesion and the disability of working with different datasets fields. So, it's desirable to develop unsupervised algorithm that learn and decode current document features rather than training on its belonging corpus features.

6. CONCLUSION AND FUTURE WORKS

This paper provides an elaborative study of different extractive text summarization techniques and especially focusing on recent efforts and advances in unsupervised approaches. Moreover, we present quick discussion on text summarization types and the evaluation task. While, there are many researchers focusing on improving extractive summarization by supervised approaches by learning datasets features, there are also other researchers work toward the improvement based on unsupervised approach. Unsupervised approaches aim to discover document hidden features or learn document semantic representation without the need to train model over datasets. Furthermore, Due to the availability issues of summaries labels, the unsupervised approaches can be used to build these labels automatically. So, there is a space to improve unsupervised techniques in extractive summarization to discover new features for documents. On the other hand, the evaluation field still representing a challenging task and need more updates as due to the variety of summarization types, it's required to find best evaluation method that works effectively with each type. Beside, while building manual summaries is a tedious task and also two human experts usually build different summaries, there are need to make evaluation method automated; but, still we don't know whether evaluation automation can be done sufficiently.

REFRENCES:

[1] N. Munot and S. S. Govilkar, "Comparative Study of Text Summarization Methods," *Int. J. Comput. Appl.*, vol. 102, no. 12, pp. 975–8887, 2014.

[2] H. P. Luhn, "The automatic creation of literature abstracts," *IBM J. Res. Dev.*, vol. 2, no. 2, pp. 159–165, 1958.

[3] K. S. Jones, "Automatic summarising: The state of the art," *Inf. Process. Manag.*, vol. 43, no. 6, pp. 1449–1481, 2007.

[4] A. Nenkova and K. McKeown, "Automatic summarization," *Found. Trends® Inf. Retr.*, vol. 5, no. 2–3, pp. 103–233, 2011.

[5] J. Goldstein, V. Mittal, J. Carbonell, and M. Kantrowitz, "Multi-document summarization by sentence extraction," in *Proceedings of the 2000 NAACL-ANLP Workshop on Automatic summarization*, 2000, pp. 40–48.

[6] R. Barzilay and K. R. McKeown, "Sentence fusion for multidocument news summarization," *Comput. Linguist.*, vol. 31, no. 3, pp. 297–328, 2005.

[7] D. M. Zajic, B. J. Dorr, and J. Lin, "Single-document and multi-document summarization techniques for email threads using sentence compression," *Inf. Process. Manag.*, vol. 44, no. 4, pp. 1600–1610, 2008.

[8] A. Nenkova, "Summarization evaluation for text and speech: issues and approaches," in *Ninth International Conference on Spoken Language Processing*, 2006.

[9] S. Gholamrezazadeh, M. A. Salehi, and B. Gholamzadeh, "A comprehensive survey on text summarization systems," in *Computer Science and its Applications, 2009. CSA'09. 2nd International Conference on*, 2009, pp. 1–6.

[10] C. T. Shubhangi, "An approach to single document text summarization and simplification," *IOSR J. Comput. Eng.*, vol. 16, no. 3, pp. 42–49, 2014.

[11] H. D. Kim, K. Ganesan, P. Sondhi, and C. Zhai, "Comprehensive review of opinion summarization," 2011.

[12] B. Liu, "Sentiment analysis and opinion mining," *Synth. Lect. Hum. Lang. Technol.*, vol. 5, no. 1, pp. 1–167, 2012.

[13] 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.

[14] D. M. Dunlavy, D. P. O'Leary, J. M. Conroy, and J. D. Schlesinger, "QCS: A system for querying, clustering and summarizing documents," *Inf. Process. Manag.*, vol. 43, no. 6, pp. 1588–1605, 2007.

[15] X. Wan, "Using only cross-document relationships for both generic and topic-focused multi-document summarizations," *Inf. Retr. Boston.*, vol. 11, no. 1, pp. 25–49, 2008.

[16] Y. Ouyang, W. Li, S. Li, and Q. Lu, "Applying regression models to query-focused multi-document summarization," *Inf. Process.*

Manag., vol. 47, no. 2, pp. 227–237, 2011.

[17] M. A. Fattah and F. Ren, "GA, MR, FFNN, PNN and GMM based models for automatic text summarization," *Comput. Speech Lang.*, vol. 23, no. 1, pp. 126–144, 2009.

[18] K. Sarkar, "Syntactic trimming of extracted sentences for improving extractive multi-document summarization," *J. Comput*, vol. 2, no. 7, pp. 177–184, 2010.

[19] 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.

[20] Y. Tao, S. Zhou, W. Lam, and J. Guan, "Towards more effective text summarization based on textual association networks," in *Semantics, Knowledge and Grid, 2008. SKG'08. Fourth International Conference on*, 2008, pp. 235–240.

[21] D. Wang, S. Zhu, T. Li, Y. Chi, and Y. Gong, "Integrating document clustering and multidocument summarization," *ACM Trans. Knowl. Discov. from Data*, vol. 5, no. 3, p. 14, 2011.

[22] C. Wang, L. Long, and L. Li, "HowNet based evaluation for Chinese text summarization," in *Natural Language Processing and Knowledge Engineering, 2008. NLP-KE'08. International Conference on*, 2008, pp. 1–6.

[23] D. Wang, T. Li, S. Zhu, and C. Ding, "Multi-document summarization via sentence-level semantic analysis and symmetric matrix factorization," in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, 2008, pp. 307–314.

[24] D. Wang, S. Zhu, T. Li, and Y. Gong, "Multi-document summarization using sentence-based topic models," in *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, 2009, pp. 297–300.

[25] J. L. Neto, A. A. Freitas, and C. A. A. Kaestner, "Automatic text summarization using a machine learning approach," in *Brazilian Symposium on Artificial Intelligence*, 2002, pp. 205–215.

[26] D. R. Radev, W. Fan, and Z. Zhang, "Webinessence: A personalized web-based multi-document summarization and recommendation system," *Ann Arbor*, vol. 1001, p. 48103, 2001.

[27] L. Agnihotri, J. R. Kender, N. Dimitrova, and J. Zimmerman, "User study for generating personalized summary profiles," in *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, 2005, pp. 1094–1097.

[28] A. Díaz and P. Gervás, "User-model based personalized summarization," *Inf. Process. Manag.*, vol. 43, no. 6, pp. 1715–1734, 2007.

[29] C. Kumar, P. Pingali, and V. Varma, "Generating personalized summaries using publicly available web documents," in *Web Intelligence and Intelligent Agent Technology, 2008. WI-IAT'08. IEEE/WIC/ACM International Conference on*, 2008, vol. 3, pp. 103–106.

[30] R. Witte, R. Krestel, and S. Bergler, "Generating update summaries for DUC 2007," in *Proceedings of the Document Understanding Conference*, 2007, pp. 1–5.

[31] L. Zhou, M. Ticrea, and E. Hovy, "Multi-document biography summarization," *arXiv Prepr. cs/0501078*, 2005.

[32] A. R. Deshpande and L. Lobo, "Text summarization using clustering technique," *Int. J. Eng. Trends Technol.*, vol. 4, no. 8, 2013.

[33] L. Vanderwende, H. Suzuki, C. Brockett, and A. Nenkova, "Beyond SumBasic: Task-focused summarization with sentence simplification and lexical expansion," *Inf. Process. Manag.*, vol. 43, no. 6, pp. 1606–1618, 2007.

[34] A. Haghighi and L. Vanderwende, "Exploring content models for multi-document summarization," in *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2009, pp. 362–370.

[35] D. R. Radev, E. Hovy, and K. McKeown, "Introduction to the special issue on summarization," *Comput. Linguist.*, vol. 28, no. 4, pp. 399–408, 2002.

[36] I. H. Witten, G. W. Paynter, E. Frank, C. Gutwin, and C. G. Nevill-Manning, "KEA: Practical automatic keyphrase extraction," in *Proceedings of the fourth ACM conference on Digital libraries*, 1999, pp. 254–255.

[37] S. P. Singh, A. Kumar, A. Mangal, and S. Singhal, "Bilingual automatic text summarization using unsupervised deep learning," in *Electrical, Electronics, and Optimization Techniques (ICEEOT), International Conference on*, 2016, pp. 1195–1200.

[38] L. H. Reeve, H. Han, S. V Nagori, J. C. Yang,

T. A. Schwimmer, and A. D. Brooks, "Concept frequency distribution in biomedical text summarization," in *Proceedings of the 15th ACM international conference on Information and knowledge management*, 2006, pp. 604–611.

[39] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Inf. Process. Manag.*, vol. 24, no. 5, pp. 513–523, 1988.

[40] S. R. ANIL KUMAR, JYOTIYADAV, "Automatic Text Summarization Using Regression Model (GA)," *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 3, no. 5, pp. 4253–4260, 2015.

[41] C.-Y. Lin and E. Hovy, "Automatic evaluation of summaries using n-gram co-occurrence statistics," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, 2003, pp. 71–78.

[42] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," *Text Summ. Branches Out*, 2004.

[43] D. R. Radev and D. Tam, "Summarization evaluation using relative utility," in *Proceedings of the twelfth international conference on Information and knowledge management*, 2003, pp. 508–511.

[44] S. Teufel and H. Van Halteren, "Evaluating information content by factoid analysis: human annotation and stability," in *Proceedings of the 2004 conference on empirical methods in natural language processing*, 2004.

[45] A. Nenkova and R. Passonneau, "Evaluating content selection in summarization: The pyramid method," in *Proceedings of the human language technology conference of the north american chapter of the association for computational linguistics: Hlt-naacl 2004*, 2004.

[46] E. Lloret and M. Palomar, "A gradual combination of features for building automatic summarisation systems," in *International Conference on Text, Speech and Dialogue*, 2009, pp. 16–23.

[47] V. McCargar, "Statistical approaches to automatic text summarization," *Bull. Assoc. Inf. Sci. Technol.*, vol. 30, no. 4, pp. 21–25, 2004.

[48] T. Mori, "Information gain ratio as term weight: the case of summarization of ir results," in *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, 2002, pp. 1–7.

[49] 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.

[50] C. Orasan, V. Pekar, and L. Hasler, "A Comparison of Summarisation Methods Based on Term Specificity Estimation.," in *LREC*, 2004.

[51] C. Orăsan, "Comparative evaluation of term-weighting methods for automatic summarization," *J. Quant. Linguist.*, vol. 16, no. 1, pp. 67–95, 2009.

[52] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Comput. networks ISDN Syst.*, vol. 30, no. 1–7, pp. 107–117, 1998.

[53] R. Mihalcea and P. Tarau, "Textrank: Bringing order into text," in *Proceedings of the 2004 conference on empirical methods in natural language processing*, 2004.

[54] 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.

[55] X. Wan and J. Xiao, "Towards a unified approach based on affinity graph to various multi-document summarizations," in *International Conference on Theory and Practice of Digital Libraries*, 2007, pp. 297–308.

[56] 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.

[57] R. V. V. M. Krishna and C. S. Reddy, "Extractive Text Summarization Using Lexical Association and Graph Based Text Analysis," in *Computational Intelligence in Data Mining—Volume 1*, Springer, 2016, pp. 261–272.

[58] V. V. M. K. Ravinuthala and S. R. Chinnam, "A Keyword Extraction Approach for Single Document Extractive Summarization Based on Topic Centrality."

[59] K.-F. Wong, M. Wu, and W. Li, "Extractive summarization using supervised and semi-supervised learning," in *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, 2008, pp. 985–992.

[60] C. Aone, M. E. Okurowski, and J. Gorlinsky, "Trainable, scalable summarization using robust NLP and machine learning," in

Proceedings of the 17th international conference on Computational linguistics-Volume 1, 1998, pp. 62–66.

[61] J. Kupiec, J. Pedersen, and F. Chen, "A trainable document summarizer," in *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, 1995, pp. 68–73.

[62] J. M. Conroy and D. P. O'leary, "Text summarization via hidden markov models," in *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, 2001, pp. 406–407.

[63] J. D. Schlesinger *et al.*, "Understanding machine performance in the context of human performance for multi-document summarization," 2002.

[64] C. Burges *et al.*, "Learning to rank using gradient descent," in *Proceedings of the 22nd international conference on Machine learning*, 2005, pp. 89–96.

[65] K. Svore, L. Vanderwende, and C. Burges, "Enhancing single-document summarization by combining RankNet and third-party sources," in *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, 2007.

[66] C. J. Burges, R. Ragno, and Q. V Le, "Learning to rank with nonsmooth cost functions," in *Advances in neural information processing systems*, 2007, pp. 193–200.

[67] S. Li, Y. Ouyang, W. Wang, and B. Sun, "Multi-document summarization using support vector regression," in *Proceedings of DUC*, 2007.

[68] M. Fuentes, E. Alfonseca, and H. Rodríguez, "Support vector machines for query-focused summarization trained and evaluated on pyramid data," in *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, 2007, pp. 57–60.

[69] L. Li, K. Zhou, G.-R. Xue, H. Zha, and Y. Yu, "Enhancing diversity, coverage and balance for summarization through structure learning," in *Proceedings of the 18th international conference on World wide web*, 2009, pp. 71–80.

[70] M. A. Fattah, "A hybrid machine learning model for multi-document summarization," *Appl. Intell.*, vol. 40, no. 4, pp. 592–600, 2014.

[71] M. Mendoza, S. Bonilla, C. Noguera, C. Cobos, and E. León, "Extractive single-document summarization based on genetic operators and guided local search," *Expert Syst. Appl.*, vol. 41, no. 9, pp. 4158–4169, 2014.

[72] D. Shen, J.-T. Sun, H. Li, Q. Yang, and Z. Chen, "Document Summarization Using Conditional Random Fields.," in *IJCAI*, 2007, vol. 7, pp. 2862–2867.

[73] P. Fung, G. Ngai, and C.-S. Cheung, "Combining optimal clustering and hidden Markov models for extractive summarization," in *Proceedings of the ACL 2003 workshop on Multilingual summarization and question answering-Volume 12*, 2003, pp. 21–28.

[74] H. J. Jain, M. S. Bewoor, and S. H. Patil, "Context Sensitive Text Summarization Using K Means Clustering Algorithm," *Int. J. Soft Comput. Eng.*, vol. 2, no. 2, 2012.

[75] A. El-Kilany and I. Saleh, "Unsupervised document summarization using clusters of dependency graph nodes," in *Intelligent Systems Design and Applications (ISDA), 2012 12th International Conference on*, 2012, pp. 557–561.

[76] A. Sharaff, H. Shrawgi, P. Arora, and A. Verma, "Document Summarization by Agglomerative nested clustering approach," in *Advances in Electronics, Communication and Computer Technology (ICAECCT), 2016 IEEE International Conference on*, 2016, pp. 187–191.

[77] S. 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.

[78] G. PadmaPriya and K. Duraiswamy, "An approach for text summarization using deep learning algorithm," *J. Comput. Sci.*, vol. 10, no. 1, pp. 1–9, 2014.

[79] M. Yousefi-Azar and L. Hamey, "Text summarization using unsupervised deep learning," *Expert Syst. Appl.*, vol. 68, pp. 93–105, 2017.

[80] W. Yin and Y. Pei, "Optimizing Sentence Modeling and Selection for Document Summarization.," in *IJCAI*, 2015, pp. 1383–1389.

[81] Z. Cao, F. Wei, S. Li, W. Li, M. Zhou, and W. Houfeng, "Learning summary prior representation for extractive summarization," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*

and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), 2015, vol. 2, pp. 829–833.

[82] Z. Cao, W. Li, S. Li, F. Wei, and Y. Li, "Attsum: Joint learning of focusing and summarization with neural attention," *arXiv Prepr. arXiv1604.00125*, 2016.

[83] R. Nallapati, F. Zhai, and B. Zhou, "SummaRuNNer: A Recurrent Neural Network Based Sequence Model for Extractive Summarization of Documents.," in *AAAI*, 2017, pp. 3075–3081.

[84] S. T. Dumais, "Improving the retrieval of information from external sources," *Behav. Res. Methods, Instruments, Comput.*, vol. 23, no. 2, pp. 229–236, 1991.

[85] J. Steinberger and K. Jezek, "Using latent semantic analysis in text summarization and summary evaluation," *Proc. ISIM*, vol. 4, pp. 93–100, 2004.

[86] G. Murray, S. Renals, and J. Carletta, "Extractive summarization of meeting recordings.," 2005.

[87] 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.

[88] R. M. Badry, A. S. Eldin, and D. S. Elzanfally, "Text Summarization within the Latent Semantic Analysis Framework: Comparative Study," *Int. J. Comput. Appl.*, vol. 81, no. 11, 2013.

[89] J.-Y. Yeh, H.-R. Ke, W.-P. Yang, and I.-H. Meng, "Text summarization using a trainable summarizer and latent semantic analysis," *Inf. Process. Manag.*, vol. 41, no. 1, pp. 75–95, 2005.

[90] S. T. Davis, J. M. Conroy, and J. D. Schlesinger, "OCCAMS--An Optimal Combinatorial Covering Algorithm for Multi-document Summarization," in *Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conference on*, 2012, pp. 454–463.

[91] H. Khosravi, E. Eslami, F. Kyoomarsi, and P. K. Dehkordy, "Optimizing text summarization based on fuzzy logic," in *Computer and Information Science*, Springer, 2008, pp. 121–130.

[92] H.-H. Huang, Y.-H. Kuo, and H.-C. Yang, "Fuzzy-rough set aided sentence extraction summarization," in *Innovative Computing, Information and Control, 2006. ICICIC'06. First International Conference on*, 2006, vol. 1, pp. 450–453.

[93] L. Suanmali, M. S. Binwahlan, and N. Salim, "Sentence features fusion for text summarization using fuzzy logic," in *Hybrid Intelligent Systems, 2009. HIS'09. Ninth International Conference on*, 2009, vol. 1, pp. 142–146.

[94] M. S. Binwahlan, N. Salim, and L. Suanmali, "Fuzzy Swarm Based Text Summarization 1," 2009.

[95] S. A. Babar and P. D. Patil, "Improving performance of text summarization," *Procedia Comput. Sci.*, vol. 46, pp. 354–363, 2015.

[96] J. Yadav and Y. K. Meena, "Use of fuzzy logic and wordnet for improving performance of extractive automatic text summarization," in Advances in Computing, Communications and Informatics (ICACCI), 2016 International Conference on, 2016, pp. 2071–2077.

[97] Y. J. Kumar, F. J. Kang, O. S. Goh, and A. Khan, "Text summarization based on classification using ANFIS," in Asian Conference on Intelligent Information and Database Systems, 2017, pp. 405–417

[98] Y. Ko and J. Seo, "An effective sentence-extraction technique using contextual information and statistical approaches for text summarization," Pattern Recognit. Lett., vol. 29, no. 9, pp. 1366–1371, 2008

Table 5: Comparison of Unsupervised Extractive Text Summarization Techniques

Year	Used Algorithm	Dataset	Evaluation	Comments
2004	Graph-based ranking algorithm (TextRank) [53]	DUC 2002	ROUGE-1 = 0.4229	**Input document.** Single document. **Adv.** Adaptability with any language or domain.
2004	Graph-based ranking algorithm (LexRank) [54]	DUC 2003, DUC 2004	**On DUC 2003,** ROUGE-1 = 0.3646 **On DUC 2004,** ROUGE-1 = 0.3966 **On 17% noisy DUC 2003,** ROUGE-1 = 0.3621 **On 17% noisy DUC 2004,** ROUGE-1 = 0.3905	**Input document.** Multi documents. **Adv.** Obtain good information coverage in generated summary. Prevents unnaturally high idf scores from increasing the score of a sentence that is unrelated to the topic (work well with noisy data).
2005	LSA+TRM [89]	100 political articles from New Taiwan Weekly	Recall = Precision = F- measure = 0.4442	**Input document.** Single document. **Adv.** Generated summary composed of semantically related sentences. Approach is language independent. **Dis-Adv.** Take large time to compute SVD. Difficult to obtain best dimension reduction. Shortage of coherence.
2006	Fuzzy-Rough set (Fuzzy c-mean clustering) [92]	8 pdf articles from Journal of Artificial Intelligence Research (JAIR)	F-Measure = 0.4620391	**Input document.** Single document. **Adv.** Give good information coverage and reduce redundancy.
2007	Graph ranking algorithm (Affinity Graph) + Greedy algorithm (for high information richness & novelty). [55]	DUC 2002, DUC 2003, DUC 2004, DUC 2005	**On DUC 2002,** ROUGE-1 = 0.38111, ROUGE-2 = 0.08163, ROUGE-W = 0.12292 **On DUC 2004,** ROUGE-1 = 0.39926, ROUGE-2 = 0.08793, ROUGE-W = 0.12228 **On DUC 2003,** ROUGE-1 = 0.36187, ROUGE-2 = 0.07114, ROUGE-W = 0.11464 **On DUC 2005,** ROUGE-1 = 0.38354, ROUGE-2 = 0.07069, ROUGE-W = 0.10080	**Input document.** Multi documents. **Adv.** Generate generic and Topic-focused summaries. Handle redundancy issue. **Dis-Adv.** Words are independent with each other; so, it may contain a shortage in semantic relations.

2009	WF+TE+CQP features in DUC 2002.
	And WF+CQP features in fairy tales. [46]
		DUC 2002, 5 articles from fairy tales domain	**On DUC 2002,** F- measure of:
			ROUGE-1 = 0.45611, ROUGE-2 = 0.20252, ROUGE-SU4 = 0.22200, ROUGE-L = 0.41382
			On fairy tales, F- measure of:
			ROUGE-1 = 0.41797, ROUGE-2 = 0.10267, ROUGE-SU4 = 0.15898, ROUGE-L = 0.33742
				Input document. Single document.
				Adv. Can summarize documents that have no title. Doesn't require much processor. Handle redundancy problem.

2009	Fuzzy-Logic [93]	DUC 2002	**ROUGE-1:**
			Precision = 0.47589,
			Recall = 0.46660,
			F-Measure = 0.47019
				Input document. Single document.
				Adv. Solve binary values of features or features that have low and high values; so, it balance features values to balance weight in computations.

2012	dependency graphs + Louvain clustering algorithm (keywords level) [75]	DUC 2001, DUC 2002, British Colombia conversation corpus (BC3), Concisus corpus of event summaries	**Recall Score %: On DUC 2001,** ROUGE-1 = 45.7, ROUGE-L = 40.6, ROUGE-SU1 = 26.2 **On DUC 2002,** ROUGE-1 = 48.8, ROUGE-L = 44, ROUGE-SU1 = 29.4 **On BC3,** ROUGE-1 = 79.8, ROUGE-L = 79.4, ROUGE-SU1 = 71.8 **On Concisus,** ROUGE-1 = 47.7, ROUGE-L = 39.1, ROUGE-SU1 = 30.6	**Input document.** Single document. **Adv.** Can summarize documents that have no title. Can summarize multiple genres of documents and is language independent.
2012	LSA + Optimization methods (Greedy method + Dynamic programming) [90]	DUC 2005, DUC 2006, DUC 2007, TAC 2008, TAC 2009, TAC 2010, TAC 2011,	**On DUC 2005,** ROUGE-2 = 0.081, ROUGE-SU4 = 0.134, **On DUC 2006,** ROUGE-2 = 0.102, ROUGE-SU4 = 0.152, **On DUC 2007,** ROUGE-2 = 0.128, ROUGE-SU4 = 0.175, **On TAC 2008,** ROUGE-2 = 0.103, ROUGE-SU4 = 0.136, **On TAC 2009,** ROUGE-2 = 0.110, ROUGE-SU4 = 0.142, **On TAC 2010,** ROUGE-2 = 0.108, ROUGE-SU4 = 0.135, **On TAC 2011,** ROUGE-2 = 0.131, ROUGE-SU4 = 0.162	**Input document.** Multi documents. **Adv.** Using greedy method and dynamic programming algorithm to handle weight terms computation task and sentences extraction task separately which achieve high coverage with low redundancy. The model is language-independent.
2013	Graph ranking algorithm + Association rule mining + Greedy algorithm (for maximum coverage & relevance). [56]	DUC 2004, 5 real life documents in news.	**On DUC 2004,** ROUGE-2: Recall = 0.093, Precision = 0.099, F-measure = 0.097 ROUGE-SU4: Recall = 0.015, Precision = 0.021, F-measure = 0.019	**Input document.** Multi documents. **Adv.** Can discover correlations between terms by association rules. A flexible and portable approach.

2014	Deep learning (Restricted Boltzmann Machine) [78]	Documents from networking and software engineering domains	**On networking domain,**
			Recall = 0.429, Precision = 0.6, F-measure = 0.490
			On software engineering domain,
			Recall = 0.342, Precision = 0.83, F-measure = 0.469
				Input document. Multi documents.
				Dis-Adv. Sensitivity to datasets.

2015	Fuzzy-Logic + LSA [95]	10 different datasets	**Average results (%):**
			Recall = 44.36375, Precision = 90.77572, F-measure = 67.56974
				Input document. Single document.
				Adv. Handle sentences scoring problem by fuzzy logic and generate semantically summaries based on LSA.

2015	Deep learning (DBN) + Dynamic programming [77]	DUC 2005, DUC 2006, DUC 2007	**On DUC 2005,** ROUGE-1 = 0.3751, ROUGE-2 = 0.0775, ROUGE-SU4 = 0.1341 **On DUC 2006,** ROUGE-1 = 0.4015, ROUGE-2 = 0.0928, ROUGE-SU4 = 0.1479 **On DUC 2007,** ROUGE-1 = 0.4295, ROUGE-2 = 0.1163, ROUGE-SU4 = 0.1685	**Input document.** Multi documents. **Adv.** First algorithm to summarize query oriented multi-documents by deep learning. Significant concepts are pushed out layer by layer efficiently. Perfect model for feature extraction.
2015	Deep learning (CNN Language model) + Cosine similarity + Optimization method (DivSelect with help of PageRank algorithm) [80]	DUC 2002, DUC 2004	**On DUC 2002,** ROUGE-1 = 0.51013, ROUGE-2 = 0.26972, ROUGE-SU4 = 0.29431 **On DUC 2004,** ROUGE-1 = 0.40907, ROUGE-2 = 0.10723, ROUGE-SU4 = 0.14969	**Input document.** Multi documents. **Adv.** Powerful model in sentence representation based on Neural network language model. Handle redundancy issue. Provide DivSelect as diversified selection method. Keep the diversity and prestige of chosen sentences to be balanced.
2015	Deep learning (CNN) + Regression model + Greedy algorithm [81]	DUC 2001, DUC 2002, DUC 2004,	**The results are (%)** **On DUC 2001,** ROUGE-1 = 35.98, ROUGE-2 = 7.89, **On DUC 2002,** ROUGE-1 = 36.63, ROUGE-2 = 8.97, **On DUC 2004,** ROUGE-1 = 38.91, ROUGE-2 = 10.07,	**Input document.** Multi documents. **Adv.** Pick up the independent features of the document which reflect it. The model able to avail all potential semantic representation aspects hidden in the text. Handle redundancy issue.
2016	Graph ranking algorithm (word order relationship for connecting vertices) + Lexical association [57]	DUC 2002	ROUGE-1 Recall = 0.48645, ROUGE-2 Recall = 0.39927,	**Input document.** Single document. **Adv.** Find keywords that represent text topic based on lexical association. Spending low time while extracting keywords. Good coherence in the final summary.
2016	Deep learning (CNN) + Greedy algorithm [82]	DUC 2005, DUC 2006, DUC 2007	**The results are (%),** **On DUC 2005,** ROUGE-1 = 37.01, ROUGE-2 = 6.99,	**Input document.** Multi documents. **Adv.** Used to summarize query-focused multi documents. Handle query relevance and saliency of sentences issues jointly together. Applied neural attention method

			On DUC 2006, ROUGE-1 = 40.90,
			ROUGE-2 = 9.40,
			On DUC 2007, ROUGE-1 = 43.92,
			ROUGE-2 = 11.55,
				simulate human nature while reading a document and having query in their mind.

2017	Deep learning (Deep Auto-encoder) [79]	Summarization and Keyword Extraction from Emails (SKE), BC3 from British Columbia University	**On Subject-oriented summarization with SKE and for 5 sentences summary length,** the ROUGE-2 Recall of LTF-ENAE (Gaussian) = 0.5031, **On Key- phrase oriented summarization with SKE and for 5 sentences summary length,** the ROUGE-2 Recall of LTF-AE = 0.5657, **On subject oriented summarization with BC3 and for 4 sentences summary length,** the ROUGE-2 Recall of LTF-AE = 0.1084,	**Input document.** Single document. **Adv.** Ability to generate concept vector representation for the original sentences. Generate high informative and semantic summaries. Handle sparse representation problem by local term frequency (LTF) and extra random noise. **Dis-Adv.** Training computational cost and the requirement of tuning the training hyper-parameters.
2017	Deep learning (GRU-RNN) + Greedy algorithm [83]	CNN/Daily Mail corpus, DUC 2002	**On Daily Mail, The Recall value with:** **75 bytes of summary length:** ROUGE-1 = 26.2, ROUGE-2 = 10.8, ROUGE-L = 14.4 **275 bytes of summary length:** ROUGE-1 = 42.0, ROUGE-2 = 16.9, ROUGE-L = 34.1 **On DUC 2002, The Recall value with 75 words of summary length:** ROUGE-1 = 46.6, ROUGE-2 = 23.1, ROUGE-L = 43.03	**Input document.** Single document. **Adv.** Interpretability of visualization for its predictions. Allow the extractive model to be trained using extractive labels (via unsupervised way which convert abstractive summaries to extractive labels), and using human (abstractive) summaries without the needs of labeled data.

2017	Adaptive Neuro-Fuzzy Inference System (ANFIS) (Fuzzy-logic based on neural network) [97]	DUC 2002	Precision = 0.7128, Recall = 0.6982,
			F-measure = 0.7054
				Input document. Single document.
				Adv. Tackle the problem of needing the human experts for building fuzzy rules by using subtractive clustering method to automatically generate rules.

2017	Graph ranking algorithm (Topic Association Graph) + Lexical association [58]	DUC 2002	**For ROUGE-1:** Precision = 0.51430, Recall = 0.61643,
			F-measure = 0.56050
			For ROUGE-2: Precision = 0.40323, Recall = 0.48410,
			F-measure = 0.43977
				Input document. Single document.
				Adv. Present new technique for connecting the vertices by the way that increase the incoming edges for topic central words (Topic Association Graph). Enabling the usage of centrality measures degrees for calculating vertices strength.