Literature Review on Extractive Text Summarization Approaches

Saiyed Saziyabegum
Ph.D. Scholar,
Sardar Patel University,
Vallabh vidhyanagar,
Gujarat, India

Priti S. Sajja, PhD
Ph.D., Professor,
G.H.Patel PG
Department of computer Science,
Sardar Patel University,
Vallabh vidhyanagar,
Gujarat, India

ABSTRACT

There is plenty of information available on internet. Important information can be considered by creating summary from available information. Manual creation of summary is complicated task. Therefore research community is developing new approaches to for automatic text summarization. Automatic text summarization system creates summary. Summary is shorter text that covers important information from original document. Summary can be created using extractive and abstractive methods. Abstractive methods are requires deep understanding of text. After understanding, it represents text into new simple notions in shorter form. Extractive approach uses linguistic and statistical approach for selection of sentences for summary. This paper presents an ample survey of recent text summarization extractive approaches developed in last few decades. Summary evaluation is also covered briefly in this paper. Finally this paper ends with conclusion of future research needed.

Keywords

Text Summarization, Extractive summarization

INTRODUCTION

In today's fast emerging world of information, text summarization [1] is very vital and required tool for understanding text information. There is a lot of text material and documents available on the internet which provides information beyond requirements and creates the situation called 'infobesity'. To select information from large amount of information from variety of sources is difficult for human beings. Due to the volume of information and unstrucutredness of information, to manually summarize information available on the internet is really challenging, complicated and difficult task.

The aim of automatic text summarization is to reduce the source text into a compact version which will preserve contents and general meaning. Advantage of Summary is that it minimizes reading time and efforts.

Types of summarization:

1.1.1 Based on processing technique: These types are based on the text processing techniques, whether text for summary is just selected based on statistical/linguistic features of sentences or the deep understanding of text is required.

1.1.1.1 Abstractive: Abstractive summarization [32][33] try to understand *the* main concepts by using in a document and represent them in basic natural language. For understanding and examining text, it uses linguistic methods. After understanding of text, first it finds the new notions and terms to best clarify it. By using these notions and terms, creating a new shorter text that can represent the most significant information from the original text document.

1.1.1.2. Extractive: Extractive summarization selects significant paragraphs, lines and words from original text and clubs them as summary. The selection of paragraph, line or word is based on statistical and linguistic features of sentences.

Extractive summaries [2] are created by extracting main text fragment (sentences or passages) based on statistical analysis of features (as word/phrase frequency, location or cue words) to locate the sentences to be extracted from the text. The content which is used most commonly or the content which is having most favorable location is considered as most important content. This approach does not require deep understanding of text.

Extractive text summarization process [31] is divided into two steps: 1) Pre Processing and 2) Processing. Pre Processing is controlled representation of the original text. It includes three activities: i) identification of Sentences boundary: Sentence boundary is identified with presence of dot at the end of sentence, in English. ii) Removal of Stop-Word: Common words with no meaning and which do not represent related information to the task are removed. iii) Stemming: the purpose of stemming is to find the stem or radix(source or origin) of each word, which give stress to its semantics. In processing step, features influencing the relevance of sentences are decided and calculated and then weights are assigned to these features using weight learning method. Total final score of each sentence is calculated using Feature-weight equation. Top ranked sentences are selected for final summary.

1.1.2 Based on purpose of processing: These types are based on the length of summary.

1.1.2.1 Indicative summaries: Indicative summaries give shortened information on the main topics of a document [4]. It gives a clear idea to reader that original document is worth reading. The typical lengths of indicative summaries range between 5 till 10% of the complete text.

1.1.2.2 Informative summaries: Informative summaries provide a replacement for full document, which retain significant details and reduce volume of information [4]. Informative summary is typically 20-30 % of the original text.

1.1.2.3 Critical or Evaluative summaries: Critical or Evaluative summaries capture the point of view of the author on a given subject. Reviews are typical example of that.

1.1.2.4 Update summaries: In Update summaries, it is considered that user have the fundamental knowledge about the topic and requires only the current updates regarding that particular topic.

1.1.3 Based on audience: These types are based on the audience who is going to use summary i.e. whether summary is for group of people, or it is for a specific user's query.

1.1.3.1 Generic summaries: Generic summary result is aimed at a wide group of people, equal important is given to all major topics.

1.1.3.2 Query-based summaries: The result is based on a question or query.

1.1.3.3 User specific or Topic specific summaries: This type of summary is customized for the concern of exacting user or highlight only particular topic.

1.1.4 Based on Number of Document(s) and language : These types are based on number of documents to summarize and languages of text document(s).

1. 1.4.1 Based on documents:

1.1.4.1.1 Single document summarization: summary is created from single document.

1.1.4.1.2 Multi document summarization: summary is created from multiple documents.

1.1.4.2 Based on language:

1.1.4.2.1 Single language summarization: summary is created from single language document.

1.1.4.2.2 Multi language summarization: summary is created to multiple language documents.

Problems with the text summarization

The Problems with extractive text summarization [46] [47] are:

1. Sentences selected for summary generally longer, so unnecessary parts of the sentences for summary also get included & they consume space.
2. If summary size is not long enough, the important information scattered in various statements cannot captured using extractive summarization.
3. Information which is clashing may not be presented accurately.
4. Sentences frequently contain pronouns. They lose their referents when used out of context. If irrelevant sentences are clubbed together, may lead to confusing understanding of anaphors which will result in erroneous representation of original information.
5. The same problem is with multidocument summarization, because extraction of text is performed on different sources. Post processing can be used to deal with these troubles, for example, replacing pronouns with their background, replacing relative temporal expression with actual dates etc.

Problems with the abstractive text summary [46] are:

The challenge for abstractive summary is the problem representation. Capability of system is dependent on how carefully problem is represented. The system cannot able to summaries the things that are not represented properly in problem.

Features For Extractive Text Summarization

Some features [2] [5][29] to be considered for including a sentence in final summary are:

a. Title word feature:
Sentences containing words that are same as title, are also pinpointing of the matter of the document. Such sentences are having higher chances to get included into summary.

b. Content word (Keyword) feature:
Content words or Keywords are generally nouns. They can be determined using term frequency - inverse document frequency. Sentences which contain keywords are of higher chances to get included into summary.

c. Sentence Length feature:
Very large and very short sentences are not considered in summary.

d. Sentence position feature:
Sentence position matters a lot in abstractive text summarization. Usually first and/or last sentence of first and/or last paragraph of a text document are additional important and are having higher chances to get included into summary.

e. Proper Noun feature:
Proper noun can be name of an entity, name of place and name of any concept etc. Sentences containing proper nouns are having higher chances to get included into summary.

f. Upper-case word feature:
Sentences containing acronyms or proper names are included in summary.

g. Cue-Phrase Feature:
Sentences containing any cue phrase are most possible to be in summaries.

h. Biased Word Feature:
If a word appearing in a sentence is from biased list of words, then that sentence is important. Biased word list is predefined and may contain domain specific words.

i. Font based feature:
Sentences containing words written in upper case, bold, italics or Underlined fonts are considered more important.

j. Pronouns: P
ronouns such as "they, it, he , she" cannot get included in summary unless they are expanded into matching nouns.

k. Presence of non-essential information:
Some words are indicators of pointless information e.g. "because", "furthermore", and "additionally", and typically occur in the beginning of a sentence." True" or "1" value can be taken for this feature if the sentence contains at least one of these words, and "false" or "0" in opposite case.

l. Sentence-to-Sentence Cohesion:
For each sentence of document, similarity between "s1" and each other sentence s' of the document is calculated. By summing up all those similarity values ,raw value of this feature can be obtained for specific sentence. The process is repeated for all sentences.

m. Sentence-to-Centroid Cohesion:

For each sentence, first compute the vector representing the centroid of the document. Centroid is the arithmetic average over the corresponding coordinate values of all the sentences of the document; then computing the similarity between the centroid and each sentence; raw value of this feature can be obtained for each sentence. The process is repeated for all sentences.

n. Discourse analysis:

Discourse level information [38], in a text is one of good feature for text summarization. In order to produce a coherent, assured summary, and to determine the flow of the author's argument, it is necessary to determine the overall discourse structure of the text and then removing sentences peripheral to the main message of the text.

2. EARLY RESEARCH ON TEXT SUMMARIZATION

The source text is analyzed with the help of program [7]. Statistical information resulting from word frequency and distribution is used by the machine for computing importance, for each word and for each sentence [7]. The frequency of a particular word in an article gives useful measure to decide its importance. After performing stemming and stop word removal, list of content words prepared which was sorted based on frequency in decreasing order, the number providing importance of the word. On a sentence level, importance was obtained by calculating the number of occurrences of significant words in a sentence, and the linear distance between them due to the presence of non-significant words. Ranks are assigned for all sentences in order of their importance, and the top ranking sentences are generally selected to form the summary.

The approach that is based on sentence position, it is a feature helpful in finding prominent parts of documents [3]. The author examined 200 paragraphs and concludes that the topic sentence came as the first one in approximate 85% of the paragraphs, and it was the last sentence in 7% of the time. Thus, these two are accurate ways to select topic sentence.

The other method [8] is a procedure for creating manual extracts, which was applied in a set of 400 technical documents. The two features of word frequency and positional importance were included from the previous two research and two other features used were namely pragmatic (cue) words: the weight of a sentence is calculated by the presence or absence of certain (pragmatic) cue words in the cue dictionary (presence of words like significant, or hardly) Title and heading words: sentence weight is calculated as a sum of all the content words those are in the title and sub title of a text. Weights were assigned to each of these features manually to calculate score of each sentence [8]. During evaluation, it was shown that about 44% of the automatic extracts matched the manual extracts.

The ANES text extraction system [10] describes a system that performs domain-independent automatic compression of news. System first calculate tf*idf weight for all term. Second it selects the term with a high tf*idf – weight & also select head line terms. Third it calculate sum of weight of all such words and find weight of each sentence. Fourth it selects sentences with high score as part of summary.

In Clustering and Building links [12] they defines the concept of serial clustering of words in text, and discovers the value of such clustering as a sign of a word bearing content. The numerical measures proposed may also be of value in assigning weights to terms in requests.

3. EXTRACTIVE SUMMARIZATION APPROACHES FOR SINGLE DOCUMENT SUMMARIZATION

Extractive summarizers [30][35] aims to selecting most important sentences in the document and also maintains a low redundancy in the summary.

3.1 Term frequency- inverse document frequency based approach

Bag-of-words model is constructed at sentence level, with the usual weighted term-frequency and inverse sentence frequency standard [16], where sentence-frequency is calculated by finding the number of sentences in the document that contain that term. These sentence vectors are then scored by similarity to the query and the highest scoring sentences are taken as part of the summary. This is a use of Information Retrieval concept. Summarization is query-specific, but can be adapted to be generic. To generate a generic summary, nonstop words that occur most frequently in the document may be taken as the query words. Since these words represent the topic of the document, they can generate generic summaries .Term frequency is usually 0 or 1.If users create query words the way they create for information retrieval, then the query based summary generation would become generic summarization.

3.2 Cluster based Approach

Cluster based summarization [17] in which the significance of narrative text classification in the task of automatic key phrase extraction in Web document corpora. They target three methods, TFIDF, KEA, and Keyterm, used to extract key phrases from all the plain text and from only the narrative text of Web pages. ANOVA tests are used to analyze the ranking data collected in a user study using quantitative measures of acceptable percentage and quality value. The assessment shows that key phrases extracted from the narrative text only are considerably better than those obtained from all plain text of Web pages. This demonstrates that narrative text classification is vital for efficient key phrase extraction in Web document corpora.

They presented cluster based approach [50] which consists of two steps. First sentences are clustered and then representative sentences are defined and extracted based on each cluster. They developed a modified discrete differential evolution algorithm to optimize the objective functions. Methods were evaluated ROUGE-1, ROUGE-2 and ROUGE-su4metrics.

3.3 Naive-Bayes Approaches

This method [9] derived from [8]. The classification function categorizes each sentence as worth to extract or not, using a naive-Bayes classifier. The features were accommodating to[8], but also included the sentence length and uppercase words. Each sentence was given a score and only the top n sentences were extracted. To evaluate the system, a corpus of technical documents along with manual summaries was used. The manual abstract was compared with actual document sentences for each sentence. It checks whether sentences are exactly matching, join of two or more statements are matching or sentences were not matching. The auto-extracts were then evaluated against this mapping. Analysis shows that a system using which is using position and the cue features, sentence length, sentence feature was giving best abstract.

There is other naive-Bayes classifier, but with more prosperous features [11]. They expressed a system named DimSum that was using features like term frequency (tf) and inverse document frequency (idf) to find the words who are showing key notions of document. The idf was computed from a large corpus of the same domain as the referenced documents. With the help of named-entity tagger, each entity was assumed as a single token. They deployed some low discourse analysis like reference to same entities in the text, preserving cohesion. The references were resolved at a very low level by connecting name aliases within a document like \U.S.A" to \United States of America". Synonyms and linguistic variations were also merged while considering lexical terms, the past being identified by using Wordnet[14]. The corpora used for testing purpose were from newswire.

3.4 Rich feature and decision trees based Approaches

In this Approach they considered the significance of a feature called sentence position[15].They weighted a sentence by its position in text, this method is known as position method, taken from the point that texts generally follow a predictable discourse structure, and important sentences occurs in certain specifiable locations (e.g. title, abstracts, etc). but, since the discourse structure drastically varies from domain to domain, the position method cannot be defined as naively as in[3] .The paper focuses on techniques of customizing the position method for different type. In this they have used newswire corpus. They used text about computer and hardware related to it, along with collection of key topic words and abstract of six lines, the authors measured the yield of each and every sentence position in opposition to the topic keywords and ranked the sentence positions by their average yield to create the Optimal Position Policy (OPP) for topic positions for the type. Two kinds of evaluation were performed. Some earlier unobserved text was used to test whether the same procedure would work in a different domain. The first evaluation showed that the outline was exactly like the training documents. In the second evaluation, word partly cover of manual abstracts with the extracted sentences was measured. Content in abstracts were compared with content on the selected sentences and equivalent precision and recall values were calculated. A high level of matching indicated the effectiveness of the position method.

In [21] leaves assumption that features are independent of each other and modeled the problem of sentence extraction by using decision trees, instead of a naive-Bayes classifier. He studied a many features and their effect on sentence extraction. Publicly available collection of text is used for this purpose, which is divided into various topics, provided by the TIPSTER-SUMMAC6 evaluations. The dataset contains necessary text fragments (phrases, clauses, and sentences) which must be included in summaries to answer some TREC topics. These fragments were evaluated by a human. The experiments described in the paper are with the SUMMARIST system. The system extracted sentences from the documents and those were matched against human extracts, like most early work on extractive summarization. Some new features were the query signature i.e. normalized score given to sentences depending on number of query words that they contain, IR signature i.e. the most salient words in the corpus, similar to the signature words[11] ,numerical data i.e. Boolean value 1 given to sentences that contained a number in them, proper name i.e. Boolean value 1 given to sentences that contained a proper name in them, pronoun or adjective i.e. Boolean value 1 given to sentences that contained a pronoun or adjective in them. It is noted that some features like the query signature are question-oriented because of the setting of the evaluation, unlike a generalized summarization framework. The author experimented with various methods, like using only the positional feature, or using a combination of all features by adding their values. When evaluated by matching machine extracted and human extracted sentences, the decision tree classifier was clearly the victor for the entire dataset, but for three topics a naive combination of features beat it. Lin guess that this is due to some of the features were independent of each other. Feature analysis suggested that the IR signature was an important feature and it confirms the early findings of Luhn [7].

3.5 Hidden markov model based approach

In this approach they [23] modeled the problem of extracting a sentence from a document using a hidden Markov model (HMM). It is a sequential model; it is used to account for local dependencies between sentences. Only three features were used: position of the sentence in the document, number of terms in the sentence, and probability of the sentence terms given the document terms. They used TREC dataset as training corpus, the authors obtained the maximum-likelihood estimate for each transition probability, forming the transition matrix called estimate matrix. Element of this matrix is the empirical probability of transitioning from state one state to other. With each state, there is an output function associated. They assume that features are multivariate normal. The output function for each state was therefore estimated by using the training data to compute the maximum probability estimate of its mean and covariance matrix. Evaluation was done by comparing with human generated extracts.

3.6 Log-linear model approach

Making use of Log-linear models could experientially show that the summarization system produced better summaries than the Naïve-Bayes model[29] .The summaries produced from such a model were evaluated, using the standard F-score. The features included word pairs, sentence length, sentence position and discourse features like inside introduction or inside conclusion.

3.7 Neural network based approach

A neural network is trained to study the applicable features of sentences that can be selected in the summary of the article[27]. The neural network is then modified to generalize and combine the relevant features visible in summary sentences. Finally, the modified neural network is used as a filter to create summaries of news articles.

3.8 Graph theoretic approach

The graph based approach is used for extracting the most relevant sentences from the original document to form a summary. The design of this approach is to make use of both the local and global properties of sentences. The local property can be considered as clusters of significant words for each sentence, while the global property can be thought of as relations of all sentences in the document. These two properties are combined to get a single measure reflecting the informativeness of sentences. The first step involved in the process of summarizing one or more documents is identifying the issues or topics addressed in the document. Graph theoretic representation [18] of paragraphs provides a method of identification of these themes. After the common preprocessing steps, i.e. stop word removal and stemming, sentences of the documents are represented as nodes in an undirected graph. Nodes represent sentences. Two sentences are connected with an edge if they share some common words. This representation yields two results: The sub-graphs that are unconnected to the other sub graphs forms dissimilar or distinct topics covered in the documents. This allows a choice to cover in summary. For query-specific summaries, it is easy to select sentences only from the pertinent sub graph, while for generic summaries; sentences may be taken from each of the sub-graphs. The second result yielded by the graph-theoretic method is the recognition of the important sentences from the document. The nodes with high cardinality i.e. number of edges connected to that node, are the important sentences in the partition, and thus it carry higher preference to be included in the summary.

3.9 Latent semantic analysis based approach

Using Singular Value Decomposition (SVD) can find principal orthogonal dimensions of multidimensional data. It is named as LSA because SVD applied to document word matrices, groups documents that are semantically related to each other, even if they do not share the common words. By using SVD in the form of Latent Semantic Indexing (LSI), Principal Component Analysis (PCA) etc. are used in text summarization systems[13]. SVD based methods identify mutually orthogonal dimensions of the sentence vectors, selecting a representative sentence from each of the dimensions ensures relevance to the document, and orthogonality ensures non-redundancy. It is to be noted that this property applies only to data that has principal dimensions inherently.

3.10 Concept obtained approach

This approach used to obtain concepts of words based on HowNet [19][20]. It uses concept as feature, instead of word. This approach uses theoretical vector space model to create a rough summarization, and then calculate degree of semantic comparison of sentence for reducing redundancy. It first uses Hownet as tool to obtain concept of text, and set up conceptual vector space model, it then calculates importance of concept based on conceptual vector space model. At last it Generate the final summary by calculating importance of sentence and reducing the redundancy of summarization.

3.11 Fuzzy logic based approach

This method considers each features of a text such as sentence length, similarity to key word and others as the input of fuzzy system[2][22]. Then, it develops and enters all the rules required for summarization, in the knowledge base of system. Then, a value between zero to one is obtained for each sentence in the output based on sentence characteristics and the available rules in the knowledge base. The obtained value as an output decides the measure of the importance of the sentence for the final summary. The input membership function for each feature is divided into three membership functions which are composed of insignificant values (low L), very low (VL), medium (M), significant values (High h) and very high (VH). The important sentences are extracted using IF-THEN rules according to the feature criteria. Text summarization based on fuzzy logic system architecture [28] design usually implicates selecting fuzzy rules and membership function. The selection of fuzzy rules and membership functions directly affect the performance of the fuzzy logic system. The fuzzy logic system consists of four components: fuzzifier, inference engine, defuzzifier, and the fuzzy knowledge base. In the fuzzifier, crisp inputs are translated into linguistic values using a membership function to be used to the input linguistic variables. After fuzzification, the inference engine look into to the rule base containing fuzzy IFTHEN rules to obtain the linguistic values. In the last step, the output linguistic variables from the inference are converted to the final crisp values by the defuzzifier using membership function for representing the final sentence score.

3.12 Genetic algorithm and mathematical regression model based Approach

In addressing the problem of improving the content selection in automatic text summarization, statistical tools can be of great help[6] . This approach made use of a trainable summarizer, which takes into account several features to generate summaries. The effect of each sentence feature on the summarization task was investigated. Then, all the features were used in combination to train the Genetic Algorithm (GA) and Mathematical Regression (MR) models to obtain a suitable combination of the feature weights. The feature parameters were used to train the Feed Forward Neural Network (FFNN), Probabilistic Neural Network (PNN) and Gaussian Mixture Model (GMM), in order to construct a text summarizer for each model. The performance of the approach was measured at several compression rates on a data corpus composed of 100 Arabic political articles and 100 English religious articles, and the results were promising.

3.13 Query-biased and structure-preserving approaches

This is novel summarization techniques [42] to perk up the usefulness of search engines. The system includes the structure of the documents, namely the sectional hierarchy, into the output summaries. Both the structural information and the content to be displayed in the summary are selected in a query-biased way. The system uses structural and linguistic information obtained from the documents both in the summarization process and in the output summaries. The system also uses natural language processing techniques for summarization purposes such as identification of phrases as better content carriers than single words.

The other method to create query-specific summaries[43] by adding structure to documents by extracting associations between their fragments. System view a document as set of interconnected text fragments. System has the following steps: First, it structure to every document, which can then be viewed as a labeled, weighted graph, called the document graph. Then, at query time, given a set of keywords, it perform keyword proximity search on the document graphs to find how the keywords are associated in the document graphs. For each document its summary is the minimum spanning tree on the corresponding document graph that contains all the keywords. The document graph is constructed as follows. First it parses the document and split it to text fragments using a delimiter. Each text fragments becomes a node in the document graph. A weighted edge is added to the document graph between two nodes if they either correspond to adjacent text fragments in the text or if they are semantically related, and the weight of an edge denotes the degree of the relationship. System considers fragments to be related if they share common words (not stop words) and the degree of relationship is calculated by an adaptation of traditional IR term weighting formulas.

3.14 Lexical chain based Approach

A linguistic analysis is used for performing the text summarization [44],where semantically related sequences were identified in the document, and several lexical chains were extracted that form a representation of the original document.

3.15 Ranking-based sentence clustering Approach

A ranking-based sentence clustering framework [53] in which a term is treated like a text object (which is independent) rather than the feature of a sentence. In theme-based summarization Clustering of sentences is very important where different topic themes are discovered and clusters are based on these themes. Clusters contain highly related sentences. Each theme cluster is based on model, depending on this model probabilities are calculated for every target object. Object can be a document or it can be a term in each cluster. In this model set of highly ranked documents and terms are used to generate a sentence. Generative probabilities are calculated for each sentence generated from each theme cluster and posterior probabilities are calculated for each sentence. Similarity between a sentence and a cluster is computed. The above two processes are repeated until sentence clusters do not change amazingly. In the end, reallocation of each sentence occurs to the cluster which is most similar to the sentence. Once sentence clusters are obtained, the summaries are produced by selecting the highest ranked sentence from the highest ranked theme cluster to lowest ranked theme cluster, then the second highest ranked sentences from theme clusters in decreasing sequence of their ranks and so on.

3.16 Hybrid approaches

3.16.1 Fuzzy logic and LSA based approach

This hybrid approach uses fuzzy logic as a summarization sub-task improved the quality of summary by a great amount. It improves the quality of summary by incorporating the latent semantic analysis into the sentence feature extracted fuzzy logic system to extract the semantic relations between concepts in the original text [41].

3.16.2 Graph and neural network based approach

This method [39] works on the sentence extraction-based text summarization task use the graph based algorithm to calculate importance of each sentence in document and most important sentences are extracted to generate document summary. It uses usage of Part of Speech disambiguation using a recurrent neural network. System recognizes the most important sentences using various shallow linguistic features; it considers degree of connectedness among the text units to minimize the poor linking sentences in the resulting summary. These extraction based text summarization methods give an indexing weight to the document terms to compute the similarity values between sentences. The process can be described in three parts: i. preprocessing: Preprocessing Parse the document and generate sentences. ii. Graph Building: This represents a sentence as a node with all its properties and methods to handle with its behavior.iii. Sentence Ranking Algorithm: The basic approach of Sentence Rank is that a document is in fact considered the more important the more other documents link to it, but those inbound links do not count equally. First of all A document ranks high in terms of Sentence Rank, if other high ranking documents link to it.

Fuzzy logic, evolutionary algorithm and cellular learning automata based Approach

This approach uses fuzzy logic system, evolutionary algorithm and cellular learning automata [49]. Once important features are extracted, they are combined in linear fashion to show importance of each sentence. Artificial bee colony algorithm and cellular learning automata is used for calculating similarity measure. An approach is used to adjust best weights of text features using particle swan optimization and genetic algorithm. It assigns fair weights to features and at last fuzzy logic system is used to perform final scoring.

4. EXTRACTIVE SUMMARIZATION APPROACHES FOR MULTIPLE LANGUAGE TEXT SUMMARIZATION

Multilingual text summarization is used to summarize the source text in different language to the target language final summary. SimFinderML [24] identifies similar parts of text by calculating similarity over multiple features. It uses two types of features, composite features, and unary features. All features are computed over primitives, syntactic, linguistic, or knowledge-based information units extracted from the sentences. Both composite and unary features are built over the primitives.

The primitives used and features computed can be set at run-time, allowing for easy experimentation with different settings, and making it easy to add new features and primitives. Support for new languages is added to the system by developing modules conforming to interfaces for text pre-processing and primitive extraction for the language, and using existing dictionary-based translation methods, or adding other language-specific translation methods.

MEAD [26] is platform for multi-lingual summarization and evaluation. MEAD implements multiple summarization algorithms such as position-based, centroid-based, largest common subsequence, and keywords. The methods for evaluating the quality of the summaries are both intrinsic (such as percent agreement, cosine similarity, and relative utility) and extrinsic (document rank for information retrieval).MEAD's architecture consists of four stages. First, documents in a cluster are converted to MEAD's internal format which is based on XML. Second, given a configuration file or command-line options, a number of features are extracted for each sentence of the cluster. Third, these extracted features are combined into a combine score for each sentence. Fourth, these scores can be further refined after considering possible cross-sentence dependencies (e.g., repeated sentences, sequential ordering, source preferences,etc.) In addition to a number of command-line utilities, MEAD provides a Perl API which lets external programs access its internal libraries.

The Naïve Bayesian Classification with the timestamp concept for text summarization [48] works on different domains like international news, politics, sports and entertainment. The length of summary and compression rate can be specified as per User's need. The timestamp provides the summary an ordered look, which attain the coherent looking summary. It is used to extracts the more relevant information from the multiple documents. The word frequency is calculated. The system is compared with the existing MEAD algorithm and gives better outputs than the MEAD algorithm. The system is better precision, recall and F-Score. The timestamp procedure is also applied on the MEAD algorithm and the results are examined with this method. The results show that the proposed method results in lesser time than the existing MEAD algorithm to execute the summarization process.

5. EXTRACTIVE SUMMARIZATION APPROACHES FOR MULTIPLE DOCUMENT TEXT SUMMARIZATION

The Multi-document summarization framework is based on event information and word embeddings. The framework [51] was developed by extending a kp centrality - single document summarization method. It involves two different strategies. I. Single layer approach combine summaries of each input document to create final summary. Ii. The waterfall approach combines summaries in cascade fashion, according to temporal sequence of documents. Event information is used in filtering stage and to improve the sentence representations. They used skip-gram model, continuous bag-of-word model and distributed representation of sentences. Event detection uses fuzzy fingerprint method. Evaluation is performed using rouge-1 and user study.

Graphsum[52] a graph-based summarizer works on collection of documents, that determines and uses association rules to represent the connections among multiple terms during the summarization process. Graph sum uses a strategy that distinguishes between positive and negative term correlations. The graph nodes, which represent combinations of two or more terms, are first ranked by means of a Page Rank. Then, the produced node ranking is used to perform the sentence selection process.

6. EXTRACTIVE SUMMARIZATION APPROACHES FOR MULTIPLE LANGUAGE AND MULTIPLE DOCUMENT TEXT SUMMARIZATION

MINDS [25] integrate multi lingual summarization and multi document summarization capabilities using a multiengine core summarization system that provides interactive document access through hypertext summaries. It produces summaries both in English and in the original language of a document. It uses core summarization engine independent of languages. A prototype core engine has been built for English, Spanish, Russian, and Japanese documents. It uses document structure analysis and word frequency analysis as core summarization techniques. Document structure analysis involves identification of language, document structure parsing(heading, subheading, section, subsection, data and graphics gets separated for document for HTML encoding) Multilingual Sentence Segmentation and text structure heuristics (it uses rules based on document to score sentences and it is main method for scoring and selecting sentences.) Word frequency analysis sorts words of document by frequency and selects some most frequent words.

7. SUMMARY EVALUATION

Summary evaluation [34][37] is important for text summarization. Approaches to evaluation are divided into extrinsic where a summary is judged according to how much it contributes to the accomplishment of a particular task, and intrinsic wherein the quality of a summary is judged directly without reference to a particular task. Evaluation of Performance of Automatic summary can be measured using precision, recall and F-score. Precision is the number of sentences found in both system and ideal summaries divided by the number of sentences in the system summary. Recall is the number of sentences found in both system and ideal summaries divided by the number of sentences in the ideal summary. F-score is a combined measure and it combines precision and recall [45].

A set of metrics called recall oriented understudy of gisting Evaluation (ROUGE) was introduced [36], it has become the standard of automatic evaluation of the summaries, and gives a score based on the similarity in the sequences of words between a human-written model summary and the machine summary.

Evaluating summaries, either manually or automatically, is a hard task. The main difficulty in evaluation comes from the impossibility of building a fair standard against which the results of the systems can be compared. Furthermore, it is also very hard to determine what a correct summary is, because there is always the possibility of a system to generate a good summary that is quite different from any human summary, used as an approximation to the correct output.

8. CONCLUSION

Text summarization is motivating field of research and it has variety of applications. The objective of this paper is to study some important information related to the past of automatic text summarization and current trends. In this paper, more focus is given to Text summarization extractive approaches and they are categorized into different categories. Summary Evaluation is also briefly covered. Due to the problem of infobesity more effective and cutting edge hybrid technique of neural network, genetic algorithm and fuzzy logic for automatic text summarization is required.

9. REFERENCES

[1] Jezek, K., & Steinberger, J. (2008). Automatic text summarization. In*Znalosti* (pp. 1-12).

[2] Kyoomarsi, F., Khosravi, H., Eslami, E., Dehkordy, P. K., & Tajoddin, A. (2008, May). Optimizing Text Summarization Based on Fuzzy Logic. In*Computer and Information Science, 2008. ICIS 08. Seventh IEEE/ACIS International Conference on* (pp. 347-352). IEEE.

[3] Baxendale, P. B. (1958). Machine-made index for technical literature: an experiment. *IBM Journal of Research and Development*, *2*(4), 354-361.

[4] Fan, W., Wallace, L., Rich, S., & Zhang, Z. (2005). Tapping into the power of text mining.

[5] Chen, F., Han, K., & Chen, G. (2002, October). An approach to sentence-selection-based text summarization. In *TENCON'02. Proceedings. 2002 IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering* (Vol. 1, pp. 489-493). IEEE.

[6] Fattah, M. A., & Ren, F. (2008). Automatic text summarization. *World Academy of Science, Engineering and Technology*, *37*, 2008.

[7] Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of research and development*, *2*(2), 159-165.

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

[9] Kupiec, J., Pedersen, J., & Chen, F. (1995, July). A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 68-73). ACM.

[10] Brandow, R., Mitze, K., & Rau, L. F. (1995). Automatic condensation of electronic publications by sentence selection. *Information Processing & Management*, *31*(5), 675-685.

[11] Okurowski, M. G. J., Aone, C., & Larsen, I. (1999). B. A trainable summarizer with knowledge acquired from robust nlp techniques. *Mani, and MT Maybury. Advances in automatic text summarization*, 4-5.

[12] Bookstein, A., Klein, S. T., & Raita, T. (1995, July). Detecting content-bearing words by serial clustering. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 319-327). ACM.

[13] Ganapathiraju, M. K. (2002). Overview of summarization methods. *Selfpaced lab in Information Retrieval*.

[14] Miller, G. A. (1995). WordNet: a lexical database for English.*Communications of the ACM*, *38*(11), 39-41.

[15] Lin, C. Y., & Hovy, E. (1997, March). Identifying topics by position. In*Proceedings of the fifth conference on Applied natural language processing*(pp. 283-290). Association for Computational Linguistics.

[16] García-Hernández, R. A., & Ledeneva, Y. (2009, February). Word sequence models for single text summarization. In *Advances in Computer-Human Interactions, 2009. ACHI'09. Second International Conferences on* (pp. 44-48). IEEE.

[17] Zhang, Y., Zincir-Heywood, N., & Milios, E. (2005, November). Narrative text classification for automatic key phrase extraction in web document corpora. In *Proceedings of the 7th annual ACM international workshop on Web information and data management* (pp. 51-58). ACM.

[18] Kruengkrai, C., & Jaruskulchai, C. (2003, October). Generic text summarization using local and global properties of sentences. In *Web Intelligence, 2003. WI 2003. Proceedings. IEEE/WIC International Conference on* (pp. 201-206). IEEE.

[19] Wang, M., Wang, X., & Xu, C. (2005, October). An approach to concept-obtained text summarization. In *IEEE International Symposium on Communications and Information Technology, 2005. ISCIT 2005.* (Vol. 2, pp. 1337-1340). IEEE.

[20] Zamanifar, A., Minaei-Bidgoli, B., & Sharifi, M. (2008, August). A new hybrid farsi text summarization technique based on term co-occurrence and conceptual property of the text. In *Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2008. SNPD'08. Ninth ACIS International Conference on* (pp. 635-639). IEEE.

[21] Lin, C. Y. (1999, November). Training a selection function for extraction. In*Proceedings of the eighth international conference on Information and knowledge management* (pp. 55-62). ACM.

[22] Suanmali, L., Binwahlan, M. S., & Salim, N. (2009, August). Sentence features fusion for text summarization using fuzzy logic. In *Hybrid Intelligent Systems, 2009. HIS'09. Ninth International Conference on* (Vol. 1, pp. 142-146). IEEE.

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

[24] Evans, D. K. (2005). *Identifying similarity in text: multi-lingual analysis for summarization* (Doctoral dissertation, Columbia University).

[25] Cowie, J., Mahesh, K., Nirenburg, S., & Zajaz, R. (1998). MINDS-Multilingual INteractive document summarization. In *Working Notes of the AAAI Spring Symposium on Intelligent Text Summarization* (pp. 131-132).

[26] Radev, D. R., Allison, T., Blair-Goldensohn, S., Blitzer, J., Celebi, A., Dimitrov, S., ... & Otterbacher, J. (2004, May). MEAD-A Platform for Multidocument Multilingual Text Summarization. In *LREC*.

[27] Kaikhah, K. (2004). Text summarization using neural networks.

[28] Suanmali, L., Salim, N., & Binwahlan, M. S. (2009). Fuzzy logic based method for improving text summarization. *arXiv preprint arXiv:0906.4690*.

[29] Osborne, M. (2002, July). Using maximum entropy for sentence extraction. In *Proceedings of the ACL-02 Workshop on Automatic Summarization-Volume 4* (pp. 1-8). Association for Computational Linguistics.

[30] Mackeown, K., Nenkova, A., Elson, D., Passonneau, R., & Hirschberg, J. (2005). A task based evaluation of multidocument system. In *SIGIR* (Vol. 5, pp. 210-217).

[31] Gupta, V., & Lehal, G. S. (2009). A survey of text mining techniques and applications. *Journal of emerging technologies in web intelligence*, *1*(1), 60-76.

[32] Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*,*22*, 457-479.

[33] Hahn, U., & Romacker, M. (2001, March). The SYNDIKATE text Knowledge base generator. In *Proceedings of the first international conference on Human language technology research* (pp. 1-6). Association for Computational Linguistics.

[34] Nenkova, A., & Passonneau, R. J. (2004, May). Evaluating Content Selection in Summarization: The Pyramid Method. In *HLT-NAACL* (Vol. 4, pp. 145-152).

[35] Berry Michael, W. (2004). Automatic Discovery of Similar Words. *Survey of Text Mining: Clustering, Classification and Retrieval", Springer Verlag, New York*, *200*, 24-43.

[36] Lin, C. Y. (2004, July). Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop* (Vol. 8).

[37] Hovy, E., Lin, C. Y., Zhou, L., & Fukumoto, J. (2006, May). Automated summarization evaluation with basic elements. In *Proceedings of the Fifth Conference on Language Resources and Evaluation (LREC 2006)* (pp. 604-611). Genoa, Italy.

[38] Chan, S. W., Lai, T. B., Gao, W. J., & T'sou, B. K. (2000, April). Mining discourse markers for Chinese textual summarization. In *Proceedings of the 2000 NAACL-ANLP Workshop on Automatic Summarization* (pp. 11-20). Association for Computational Linguistics.

[39] Patil, A., Pharande, K., Nale, D., & Agrawal, R. (2015). Automatic text summarization. *International Journal of Computer Applications*, *109*(17).

[40] Hachey, B. (2009, August). Multi-document summarisation using generic relation extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1* (pp. 420-429). Association for Computational Linguistics.

[41] Babar, S. A., & Patil, P. D. (2015). Improving Performance of Text Summarization. *Procedia Computer Science*, *46*, 354-363.

[42] Pembe, F. C., & Güngör, T. (2007, June). Automated Querybiased and Structure-preserving Text Summarization on Web Documents. In*Proceedings of the International Symposium on Innovations in Intelligent Systems and Applications, İstanbul*.

[43] Varadarajan, R., & Hristidis, V. (2005, October). Structure-based query-specific document summarization. In *Proceedings of the 14th ACM international conference on Information and knowledge management* (pp. 231-232). ACM.

[44] Barzilay, R., & Elhadad, M. Using Lexical Chains for Text Summarization.

[45] Steinberger, J., & Ježek, K. (2012). Evaluation measures for text summarization. *Computing and Informatics*, *28*(2), 251-275.

[46] Lin, J., Ozsu, M., & Liu, L. (2009). Summarization. Encyclopedia of Database Systems.

[47] Cheung, J. C. (2008). *Comparing Abstractive and Extractive Summarization of Evaluative Text: Controversiality and Content Selection* (Doctoral dissertation, UNIVERSITY OF BRITISH COLUMBIA).

[48] Ramanujam, N., & Kaliappan, M. (2016). An Automatic Multidocument Text Summarization Approach Based on Naïve Bayesian Classifier Using Timestamp Strategy. *The Scientific World Journal*, *2016*.

[49] Abbasi-ghalehtaki, R., Khotanlou, H., & Esmaeilpour, M. (2016). Fuzzy evolutionary cellular learning automata model for text summarization. *Swarm and Evolutionary Computation*.

[50] Alguliev, R., & Aliguliyev, R. (2009). Evolutionary algorithm for extractive text summarization. *Intelligent Information Management*, *1*(02), 128.

[51] Marujo, L., Ling, W., Ribeiro, R., Gershman, A., Carbonell, J., de Matos, D. M., & Neto, J. P. (2016). Exploring events and distributed representations of text in multi-document summarization. *Knowledge-Based Systems*, *94*, 33-42.

[52] Baralis, E., Cagliero, L., Mahoto, N., & Fiori, A. (2013). GRAPHSUM: Discovering correlations among multiple terms for graph-based summarization. *Information Sciences*, *249*, 96-109.

[53] Yang, L., Cai, X., Zhang, Y., & Shi, P. (2014). Enhancing sentence-level clustering with ranking-based clustering framework for theme-based summarization. *Information sciences*, *260*, 37-50.