**IOSR Journal of Engineering (IOSR JEN)** **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
PP 01-04

# A Survey of Extractive Text Summarization Techniques for Indian Languages

Sheryl Rodrigues[#1], Sonia Fernandes[#2], Anusha Pai[#3]
*Department of Information Technology, Padre Conceicao College of Engineering, Goa,India*
sherylrodrigues94@gmail.com
fdezsonia@gmail.com
anusha.pai@gmail.com

***Abstract:*** *Automatic Text Summarization is the process of getting a condensed form of a textual document. With the growing information over the world wide web,text summarization becomes an important tool to reduce the information overload problem. Various text summarization techniques have been proposed for foreign languages but few techniques have been found for Indian languages. This paper provides a survey of extractive text summarization techniques for some Indian regional languages.*

***Keywords:*** *text summarization, extraction, abstraction, feature selection, summary generation*

## I. Introduction

Text summarizationis a technique of obtaining the gist of an entire document. Summarization allows the readers to quickly and easily understand the content of the original document without the need to read the entire document. It can be used in everyday life such as review of a book or movie, generating headlines.

Text summarization can be of two types [1], Automatic text summarization and manual text summarization. In automatic text summarization, a summary is generated by a computer program. This technique reduces redundancy of the original text. Manual text summarization is carried out by human experts. This technique is quite difficult when summarizing a large document as the human expert needs to learn and understand the document.

There are two approaches [2] for text summarization: extraction and abstraction. In extractive text summarization, certain sentences from the document are extracted based on features such as sentence position, sentence length, and only those sentences appear in the final summary. Extractive summarization may not yield a meaningful summary. Whereas,in abstractive text summarization, new sentences are generated by linguistic interpretation of the text and, hence, is quite challenging.

Indian languages can be categorized [3] as Indo-Aryan and Dravidian languages. Hindi, Marathi, Konkani, Gujarati, Punjabi, Bengali, Odia, Sindhi are some of the Indo-Aryan languages. Dravidian languages include Malayalam, Tamil, Telugu, Kannada etc.

## II. Extractive Based Summarization Techniques

Extractive based summarization techniques can be categorized [17] as Statistical method, Linguistic method and Hybrid method.

### *1. Statistical method:*

These techniques are simple. They depend on non-linguistic features of the document to extract the sentences. Features such as position, length of a sentence, term frequency(TF) and inverse document frequency(IDF), word occurrences in the document. This method extracts sentences from the document without considering the semantics of the sentences.

### *2. Linguistic method:*

This method, detectskeywords and extracts sentences based on the linguistic features of the words. It identifies the semantics of the words through part of speech tagging, lexical analysis, syntactic analysis and thesaurus usage. Linguistic methods yield a better summary as compared to statistical methods.

### *3. Hybrid method:*

This method uses a combination of both statistical and linguistic techniques.

## III. Literature Review

### *A. Malayalam*

Renjith S R et al. [4], proposed an extractive text summarization method for a single Malayalam document. This method consisted of five phases: Preprocessing, Sentence scoring, Finding similarity between sentences, Sentence ranking phase and ranking phase and Summary generation phase. Preprocessing of the input text comprised of three steps: Tokenization and POS tagging, Stop word removal and Stemming. In the sentence scoring phase, scores were assigned to sentences based on sentence position, sentence length, TF-ISF, and topic similarity. Similarity between sentences was calculated using a similarity formula. Sentences were ranked by applying the PageRank formula and finally the top k ranked sentences were selected to appear in the final summary.

### *B. Odia*

R.C.Balabantaray et al. [5], presented a text summarization system for documents written in Odia. The system first reads an Odia document and splits it into tokens.The stop words are then removed followed by stemming by an Odia stemmer. A weight is assigned to each term based on the frequency of each term in the document. The sentences are then ranked according to the weights of the individual terms in each sentence. Finally, extract the top ranked sentences.

### *C. Bengali*

Kamal Sarkar [6], proposed a Bengali text summarization system that has three steps:preprocessing, sentence ranking and summary generation. The preprocessing step includes stop word removal, stemming and segmenting the document into a collection of sentences. Sentences are ranked according to thematic term, sentence position, sentence length. Thematic terms are terms whose TF-IDF values are greater than a predefined threshold. In the summary generation phase, the top k ranked sentences are selected to appear in the summary. Value of k is decided by the user.

### *D. Punjabi*

Vishal Gupta et al. [7], proposed an approach towards feature selection for Punjabi text summarization. Sentences are selected based on features such as sentence length, keyword selection feature, number feature, sentence headline feature, Punjabi noun feature, English-Punjabi noun feature, Punjabi proper noun feature, cue phrase feature, title keywords feature. In sentence length feature, lengthy sentences are considered to be more informative and are selected. In keyword selection phase, terms having a high TF-ISF score are selected. In number feature, sentences containing numbers are considered important. In Punjabi noun feature, sentences containing Punjabi nouns are selected. In English-Punjabi noun feature, sentences containing English nouns are selected. In proper noun feature, sentences containing proper nouns are selected. In cue phrase feature, sentences containing cue phrases are selected. In title keywords feature, sentences containing title keywords are selected.

### *E. Kannada*

Jagadish S Kallimani et al. [8],proposed a text summarization method for Kannada. The method was proposed based on the working of a summarization tool, AutoSum. The process begins by reading a text article which is in utf-8 format. In the keyword extraction phase, keywords are extracted by tagging and parsing a text or by using a lexicon. Sentences are scored by using parameters such as first line, position, numerical values and keywords.Summary is generated by selecting the ranked sentences.

### *F. Hindi*

Chetana Thaokar et al. [9], proposed a model for summarizing Hindi text. This model had three steps: Preprocessing, Feature extraction and Genetic algorithm for ranking the sentences.Preprocessing step involved sentence segmentation, sentence tokenization, stop word removal and stemming. The features used in feature extraction step were average TF-ISF, sentence length, sentence position, numerical data, title feature, SOV qualification, and subject similarity. In Genetic algorithm, parameters like initial population, fitness function, selection, crossover and mutation is used. The fittest chromosome is selected. In sentence ranking, the distance between sentence score and fittest chromosome is evaluated. Sentences are then sorted and extracted on the basis of the compression rate. The generated summary was evaluated by using parameters such as precision and recall.

### *G. Tamil*

M. Hanumanthappa et al. [10], introduced a method for extracting keywords from a Dravidian language like Tamil. The method first began by tokenizing the document, followed by stop word removal to obtain vocabulary words. The vocabulary words were stored in a matrix called as Vector space model. The TF-IDF for each word was calculated and only those words having a high TF-IDF score were selected. The corresponding line number or paragraph number or page number was also extracted in this method.

***H. Marathi***

Shubham Bhosale et al. [11], proposed an algorithm for extracting keywords from Marathi e-newspapers. The system is made up of two modules: Word extraction module and Summarization module. In word extraction module, an e-newspaper article is taken as input.The article is tokenized and stop words are removed and what remains are the keywords which are then ranked. In summarization module, all the sentences containing the keywords are selected and ranked. Display only the top ranked sentences which will be 30%-40% of the original article.

***I. Telugu***

M. Humera Khanam et al. [12],presented a summarization technique to summarize a Telugu document by using Frequency based approach. The input document was first tokenized. Stop words like adverbs and conjunctions were removed. The frequency for all the remaining words were calculated. Sentences containing words of high frequency count were extracted and displayed in the final summary.

***J.Assamese***

Chandan Kalita et al.[13], proposed a text summarization technique to summarize Assamese documents. The three main steps are Preprocessing, Estimating the number of clusters and Summary generation. In preprocessing step, semantic similarity between two words is calculated by using Assamese wordnet. Similarly the form similarity and semantic similarity between two sentences is calculated by using a similarity function. In the next step, the number of clusters are determined based on the number of topics in the document. In the summary generation step, K-means algorithm is used to cluster the sentences. The central sentences from each cluster are selected. Calculate the similarity between the sentences which are not selected and the title of the document. Extract the sentences that are similar to the title. The sentences are sorted according to the occurrence in the input document and displayed in the summary.

***K. Sanskrit***

Siddhi Barve et al. [14], proposed a query based summarization technique for Sanskrit. Preprocessing module, Sentence extraction and ranking are the main phases of this system. In preprocessing module, very short and long sentences are eliminated from the document. The document is then tokenized followed by stop word removal. Morphological analysis is performed on the words and each word is resolved as a compound or sandhi. In sentence extraction, three methods are used to extract out the sentences. They are Average TF-ISF, vector space model and Graph based approach using PageRank. These three methods are used along with a query, to extract sentences that match the query. Query can be considered to be any non-stop word. Finally, the top ranked sentences are selected for generating the summary. After evaluation, it was found that vector space model and PageRank perform better as compared to average TF-ISF.

***L. Urdu***

Aqil Burney et al. [15], presented an add-in for the “Auto Summarize Tool” of MS Word. The add-in provided a way to summarize Urdu documents by using the Sentence weight algorithm. Each sentence was given a weight to decide whether it has to be included in the summary. The algorithm begins by calculating the total number of words in the document. All the stop words in the document are identified. Content words are obtained by subtracting the stop words from the total words. Weight of each sentence is calculated by dividing content words by total words. Sort the sentences in descending order according to their weights. Extract out the top ranked sentences and sort them again based on their appearance in the original document to get the summary. The summary generated by this summarizer was well formed and easy to understand.

***M.Gujarati***

JikitshaSheth et al. [16], introduced a text summarizer for Gujarati documents called Saaraansh. The main components of this summarizer are sentence extractor, word extractor, stop word identifier, Dhiya stemmer, GujStringSimilarity module, stem weightage module, LexRank module and Anaphora Resolver. Sentences are extracted and each sentence is assigned an id. Sentences are tokenized and stop words are identified and removed GujStringSimilarity module finds the lexical similarity between two words. The words are normalized to a common form and are then processed by a Gujarati stemmer called Dhiya. In stem weightage module, TF-IDF for each term is calculated. In LexRank module, cosine similarity between two sentences is calculated. A graph is constructed where an edge between two nodes describes similarity. Using the LexRank algorithm the sentences in the graph are ranked and extracted to be a part of the summary. Finally, anaphora resolution is performed by comparing the summarized text and the original document. Thus improving the sentence selection in the summary.

## IV. Conclusion

Text summarization is very useful for generating a compressed form of a document in a stipulated amount of time with least redundancy. This survey paper discusses various extractive summarization techniques for Indian regional languages. Extractive summaries pick out the most relevant sentences from the document by maintaining a low redundancy. Although various automatic text summarization systems are available for most of the commonly used natural languages for English and other foreign languages, but when it comes to Indian languages, automatic text summarization systems are still lacking. It is hoped that this work helps the new researchers to get a better understanding of extractive text summarization techniques.

## References

[1]. Richa Sharma, Prachi Sharma, "A Survey of Extractive Text Summarization", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 6, 2016

[2]. Mehdi Allahyari,SeyedaminPouriyeh, Mehdi Assefi, SaeidSafaei, Elizabeth D.Trippe, Juan B.Gutierrez, KrysKochut, "Text Summarization Techniques: A Brief Survey", arXiv, 2017

[3]. Dhanya P.M, Jathavedan M, "Comparative Study of Text Summarization in Indian Languages", International Journal of Computer Application, Volume 75-No.6, 2013

[4]. Renjith S R , Sony P, " An Automatic Text Summariztion For Malayalam Using Sentence Extraction", International Journal of Advanced Computational Engineering and Networking, Volume 3, Issue 8, 2015

[5]. R.C. Balabantaray, B.Sahoo, D.K. Sahoo, M.Swain, "Odia Text Summarization using Stemmer", International Journal of Applied Information systems(IJAIS), Volume 1-No.3, 2012

[6]. Kamal Sarkar ,"Bengali Text Summarization By Sentence Extraction", arXiv,2012

[7]. Vishal Gupta, Gurpreet Singh Lehal , "Features Selection and Weight learning for Punjabi Text Summarization, International Journal of Engineering Trends and Technology, Volume 2, Issue 2, 2011

[8]. *Jagadish S Kallimani, Srinivasa K G, Eswara Reddy B, "Information Retrieval by Text Summarization for an Indian Regional Language", IEEE 2010*

[9]. Chetana Thaokar, Dr. Latesh Malik, "Test Model for Summarizing Hindi Text using Extraction Method", IEEE Conference on Information and Communication Technologies, 2013

[10]. M. Hanumanthappa, M. Narayana Swamy , N M Jyothi, "Automatic Keyword Extraction from Dravidian Language", International Journal of Innovative Science, Engineering and Technology, Vol 1,Issue 8, 2014

[11]. Shubham Bhosale, Diksha Joshi, VrushaliBhise, Rushali A. Deshmukh, "Marathi e-Newspaper Text Summarization Using Automatic Keyword Extraction Technique", International Journal of Advance Engineering and Research Developmemt, Volume 5, Issue 3, 2018

[12]. M.Humera.Khanam, S.Sravani, "Text Summarization for Telugu Document", IOSR-Journal of Computer Engineering, Volume 18, Issue 6, 2016

[13]. Chandan Kalita, NavanathSaharia, Utpal Sharma, "An Extractive Approach of Text Summarization of Assamese using WordNet", Proceedings of ACL,2010

[14]. Siddhi Barve, Shaba Desai, Razia Sardinha, "Query –Based Extractive Text Summarization for Sanskrit",Proceedings of the 4th International Conference on Frontiers in Intelligent Computing: Theory and Applications (FICTA), 2015

[15]. Aqil Burney, Badar Sami, Nadeem Mahmood, Zain Abbas,Kashif Rizwan, International Journal of Computer Applications Volume 46– No.19, 2012

[16]. JikitshaSheth, Bankim Patel," Saaraansh: Gujarati Text Summarization System", International Journal of Computer Science and Information Technology & Security (IJCSITS), Vol.7, No.3, 2017

[17]. Santosh Kumar Bharti, Korra Sathya Babu, Sanjay Kumar Jena,"Automatic Keyword Extraction for Text Summarization: A Survey",arXiv, 2017