Automatic Text Summarization Based on Lexical-Semantic and Similarity Based On Sentence Embedding Model

Kalpana Metre
Department of Computer Engineering
BKC COE
Nashik, India
kvmetre@gmail.com

Shraddha Pawar
Department of Computer Engineering
BKC COE
Nashik, India
shraddhap113@gmail.com

Abstract— **The internet is made of online pages, news stories, status updates, blogs and much more. It is tough to browse through this material since it is unorganised and frequently discursive. Manual summary of big amounts of texts is arduous and mistake prone. Also, the findings in such form of summary may lead to diverse outcomes for a specific text. Thus, Automatic text summarization has become crucial owing to the massive rise of information and data. It identifies the most informative portion of text and creates summaries that highlight the primary objective of the provided content. It generates summary provided by summarising method which helps users to grasp the content of document instead for reading each and every individual document. So, the general purpose of Text Summarizer is to deliver the meaning of text in fewer words and phrases.**

Keywords—***Text summarization, Pre-processing, Stemming, Indian Languages, Extractive summarization***

I. INTRODUCTION

Text summary is mostly used in our work to assist readers in saving time and effort while reading lengthy papers in search of important information, in contrast to other alternatives. Text summarising is the process of condensing a lengthy material into a concise, accurate, and consistent summary. Automated summaries are in high demand since it is impossible to hand write summaries of all the content. Summarization aids users in a variety of ways, including lowering reading time, speeding up document selection, and enhancing indexing. When compared to human summaries, automatic summaries are less skewed. Summarization programs and systems are commercially in demand by the military, universities, research institutes, law firms, etc.

Text summarization is a method for extracting as much information as possible from a text by reducing it to an abstract form. If you're looking for a quick overview of a document, this tool may help you achieve just that. When summarising documents, it gives readers an overview of what they've read without having to read each one individually. As a result, text summarizer's primary goal is to condense the amount of content into shorter, more digestible chunks. There are two types of summarization systems: those that use abstractions and those that use extractions.

Extractive summaries are summaries in which sentences are extracted sequentially from the original material. Input text is subjected to statistical and linguistic analysis in order to extract the relevant sentences. The amount of material that can be extracted is limited. The phrases and sentences have been arranged according to when they were written down. Summaries of abstractive texts, on the other hand, are created by putting into practise the principles of natural language comprehension.

Summarizers of this kind often include phrases that don't appear in the original material. With the goal of better portraying an existing notion in the original article, it attempts to mimic human techniques. A good summation tool, but one that is tough to deploy.

II. LITERATURE SURVEY

There is always a need for innovative methods to assist solve challenges [6-7]. There are a wide range of ways that may be used to automatically summarise a piece of text. Extractive text summarization is the most common method used here. According to [9], shallow features for text unit scoring and summarising the highest-scoring units are used in [9].

Automated summaries that are as similar to human-generated summaries as feasible and boost not just the coverage but also the accuracy by grouping phrases to determine the primary themes in the source material. [1]

Domain summaries are created using an ontology-based methodology. It is necessary to feed the system a dataset in advance, and the system then uses this information to build summaries that include the appropriate content [2]. Random forest classification and feature scoring are used in the rule-based technique. The user-defined restrictions determine the score. For example, the rules may be established by employing verbs and nouns connected to each other; keywords and syntactic restrictions; domain constraints; and so on. [3]. Data structures are used to represent language in a graph-based approach. With directed edges, the structure of sentences may be deduced for each word unit. The connection between any two words is represented by these edges. [4].

Information The smallest unit of information in a sentence is referred to as an item based approach. This function identifies text entries, their properties, and their predicates. This is similar to Extractive Text Summarization Methods[5]. For summarization, we use similarity measures and TextRank [8]. With the help of TextRank, we were able to find some interesting findings by analysing the distribution of sentence scores and thematic similarities.

III. PROPOSED METHODODOGY

A. Architecure

Firstly Input Dataset and Preprocessing of Dataset is been done.Feature Extraction(TF/IDF) and Feature Selection is performed and given to LDA. A paper is generated by iteratively picking subjects and phrases using a probabilistic generative model based on LDA. As the name implies, Euclidean distance measures the average distance between any two places. Clustering issues often use this metric. Clustering is used to create groups of things that are similar to one another. In order to speed up the processing of enormous volumes of papers, new technologies have been created.

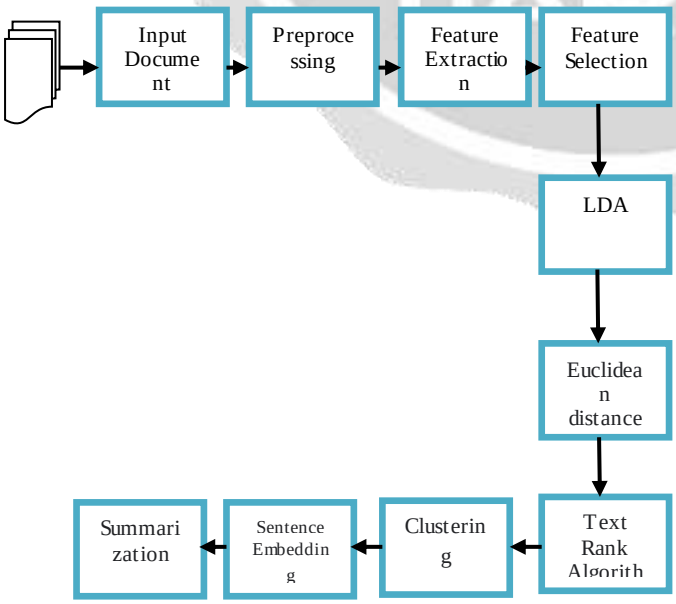

Fig1: Proposed system architecture

1) Pre-processing of input document

Pre-processing stage is essential in text summarization. It results into pre-processed data, which is ideally fit for processing stage. In general pre-processing stage consists of steps to sentence segmentation, tokenization, stop word removal, stemming, etc

i) Sentence segmentation

Sentence Segmentation is the process of breaking down/segmentation the given text document into sentences al. In this system sentence is segmented by identifying the boundary of sentence which ends with period symbol (.), question mark (?), exclamatory mark (!) and the total number of sentences present in the document are also identified.

ii) Tokenization of segmented sentences

Tokenization is the process of breaking down the sentences into words. Tokenization is done by identifying the spaces ( ), comma (,) and special symbols between the words. In this process frequency of each word is calculated and stored for further processing.

iii) Stop word removal from the list of words

Stop words are the words that do carry as important meaning as by keywords. These words are identified by supplying a list of words with less importance to the system. The system compares these stop words with the tokenized words obtained from previous phase. These stop words are then disposed as they can interfere and influence the summary that will be generated at the end.

iv) Stemming

A word can be found in different forms in the same document. These words have to be converted to their root form for simplicity. This process is known as Stemming. An algorithm is used to transform words to their root forms. In this system, Porter's stemmer method is used to turn a word into its root form using a predefined suffix list. Finally, frequency of each is word is calculated a retained for next phase.

2) Feature extraction and Feature Selection

The features like SOV (Subject Object Verb - Experimental) verification, sentence positional value (POS tagging), TF-ISF (Term Frequency/ Inverse Sentence Frequency) or TF-IDF (Term Frequency/ Inverse Document Frequency) are extracted from pre-processed sentences. Sentences are further ranked on basis of features extracted. To summarize the text, feature extraction is an essential part which requires a lot of processing with text. Feature extraction is mainly used to find the relevant or important sentence in the document. introduces 18 text features which requires the processing of named entity recognition, semantic analysis, sentiment analysis, cue phrases recognition etc. in natural language text.

3) LDA

LDA is a generative unsupervised probabilistic algorithm that isolates the top *K* topics in a data set as described by the most relevant *N* keywords. In other words, the documents in the data set are represented as random mixtures of latent topics, where each topic is characterized by a Dirichlet distribution over a fixed vocabulary. "Latent" means that topics have to be inferred rather than directly observed.

The algorithm is defined as a generative model, which means that it relies on some a priori statistical assumptions, i.e.:

- Word order in documents is not important.

- Document order in the data set is not important.
- The number of topics has to be known in advance.
- The same word can belong to multiple topics.
- Each document in the total collection of *D* documents is seen as a mixture of *K* latent topics.
- Each topic has a multinomial distribution over a vocabulary of words *w*.

4) Euclidean distance

The basis of many measures of similarity and dissimilarity is euclidean distance. The distance between vectors X and Y is defined as follows:

$$d(x,y) = \sqrt{\sum_{i}^{n} (x_i - y_i)^2}$$

In other words, euclidean distance is the square root of the sum of squared differences between corresponding elements of the two vectors. Note that the formula treats the values of X and Y seriously: no adjustment is made for differences in scale. Euclidean distance is only appropriate for data measured on the same scale.

5) Clustering

The clustering algorithm starts by specifying the number of final clusters, i.e. the parameter K. In each iteration, those two clusters that are the most similar (or the nearest) are merged and the number of clusters reduces by one. The similarity (or distance) between two clusters is computed by averaging over all similarity (or distance) values between each sentence of the first cluster and each sentence of the second one. The clustering algorithm proceeds until the number of clusters reaches K.

B. Algorithm: TextRank Algorithm

Input: I/P Marathi Document (D)

Output: Summary (S)

1. The first step would be to concatenate all the text included in the article.

2. Then break the material into distinct phrases

3. For each s in sentences

   • Identify each and every sentence's vector representation (word embeddings)

   • Analyze sentences for resemblances. End for

4. Once the matrix is put into graph form, sentences are vertices and similarity scores are edges, and the rank of each phrase is calculated.

In the end, a few of the most frequently-quoted words serve as the ultimate summation.

IV. CONCLUSIONS

Because of the massive rise in the quantity of material available online, a quick and efficient automated summarising method is required. Feature extraction, scoring, and graph construction are the most significant processes in this system method. For example, it might be used for search engine optimization (SEO), news clustering in Maharashtrian (Marathi), question generation, and a host of other purposes.

REFERENCES

[1] Rene Arnulfo Garcia-Hernandez., 2020. "Extractive Automatic Text summarization based on Lexical-Semantic Keywords." Autonomous University of Mexico State, Toluca 50000,Mexico.

[2] Lee, C. S., Jian, Z. W., and Huang, L. K. (2005). "A fuzzy ontology and its application to news summarization." IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 35(5), 859-880

[3] John, A., and Wilscy, M. (2013, December). Random forest classifier based multi-document summarization system. In Intelligent Computational Systems (RAICS), 2013 IEEE Recent Advances in (pp. 31-36). IEEE.

[4] Ganesan, K., Zhai, C., & Han, J. (2010, August). "Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions." In Proceedings of the 23rd international conference on computational linguistics (pp. 340-348). Association for Computational Linguistics.

[5] Genest, P. E., and Lapalme, G. (2011, June). "Framework for abstractive summarization using text-to-text generation." In Proceedings of the Workshop on Monolingual Text-ToText Generation (pp. 64-73). Association for ComputationalLinguistics.

[6] A. Sahba, J. Prevost Hypercube Based Clusters in Cloud Computing presented at 11th International Symposium on Intelligent Automation and Control, World Automation Congress 2016, Puerto Rico, July 2016

[7] A. Sahba, R. Sahba, and W.-M. Lin, "Improving IPC in Simultaneous Multi-Threading (SMT) Processors by Capping IQ Utilization According to Dispatched Memory Instructions," presented at the 2014 World Automation Congress, Waikoloa Village, HI, 2014.

[8] "Variations of the Similarity Function of TextRank for Automated Summarization", Federico Barrios, Federico Lopez, Luis Argerich, Rosita Wachenchauzer, arXiv, 2017.

[9] Hamzah Noori Fejer and Nazlia Omar, Automatic Multi-Document Arabic Text Summarization Using Clustering and Keyphrase Extraction ICIMU IEEE 2014 International Conference,978-1-4799-5423-0.