News Article Summarization using Extractive Method

Chaitanya Kale[1], Swapnil Tawhare[2], Dipti Surve[3]
B.E Computer Enginering, B.E Computer Enginnering, B.E Computer Enginnering
Department of Computer Engineering,
[1]Pillai HOC College of Engineering & Technology, Rasayani, India

Abstract: Text summarization is research field which helps to find out detailed but short information from documents which are often large in size, the documents can be from different fields such as finance, news and media, academics, politics, etc. Automatic Text summarization helps people to get more comprehensive information about the document. In other word, the process from which condensed form of document is created which tries to maintain information without losing or reducing the general meaning of the source document. The main goal is often to maintain the remarkable information. Automatic Text summarization is an important mean by using which large information can be concluded into shorter text in less amount of time and minimal effort. Thus, making it an important field of active research. Approaches of Text summarization are classified into two categories: Extractive and Abstractive. Extractive summarization techniques produce summaries by using words that are present in the document itself. Abstractive model takes a lot of time for training the machine learning model, it involves deep neural network to train the model then and requires large amount of corpus. Collecting such a large amount of corpus (i.e. around 4 to 5 gb minimum) and training time (i.e. about 800 hours minimum) is hard and tedious. This paper focuses on extractive model like Lex Rank and LSA. Using Lex Rank and LSA the big news articles are summarized in order to generate a shorter news article, which is enough for reader to make sense and to get complete idea.

Index Terms **- Extractive, Lex Rank, LSA (Latent Semantic Analysis), Stop words, Summarization.**

I. INTRODUCTION

Text summarization plays a vital role day today life. Because of continuing growth of content on world wide web and online text articles collections makes a large volume of information available to end users. The massive information either leads to wastage of significant time in browsing information or else useful information may missed out. The text summarization technology is maturing and may provide a solution for the information overload problem. Text summarization involves the process which can automatically generate a compressed version which is a small paragraph of a given text that is useful information to users. Text summarization is a complicated task which ideally would involve deep natural language processing (NLP) capacities [1]. In order to simplify issue, the method Extractive [2] is going to be use Lex Rank [3] and LSA which are type of extractive algorithms. By implementing text summarization, it saves time to search or to get to conclusion of the article. Big companies like Google, Amazon are use text summarization for providing better relevance result to the user. For example, Google assistant, Amazon Alexa. Extractive Summarization produces condense form to the original documents which helps in retrieval of necessary information from the huge volumes of text documents and identification of context of document [4]. Relaying on summary the user was provided with a facility to find the most desirable documents. Because of multiple irrelevant documents to a query there is a necessity of document summarization which summarizes the documents based on analysis of text, ranking them according to their values and generates new sentences where these sentences may not directly contain the actual keyword but conceptually related to the word that is used in search.

II. LITERATURE SURVEY

Data is growing in massive amount on internet and time plays very important role in every person's life. It is impossible to read whole data to a single person daily. So, to summarize the data [5] had make summarization base on Lexical chains using Silber and McCoy approach which basically does is lexical chain pre-process the data and find relation between the words that can be term of synonyms or identities. For example, two nouns need not to be identical but are use in the same sense i.e. are synonyms (The bike is red. My motorcycle is blue). There method works as it takes input pre-process it after that by tokenize, pronoun resolution it filters the noun and by using lexical chain generator it extracts the sentence and the summary is generated. The problem with this paper is every time using this method the first sentence scores higher.

To overcome the problem [6] used different methods which scores on base on word frequency, upper case, numeric data, sentence length and sentence position. Word frequency in this it removes all stop-word, calculate the number of distinct words creating simple structure given to the word to the number of times it appears in the sentence, then after for each sentence it adds up the word frequency to score each word in sentence [7]. Upper Case removes all stop words [8] before counting the capital letters in the sentence using the formula. Whereas Sentence Length calculate the large sentence length and score them

according to it. Sentence Position are ranked on the base of text and sentence score [9]. Then after combining all ranking services it gives output. It also uses google translator to convert sentence into any language.

III. EXISTING SYSTEM

Existing system of text summarizer uses no logical approach. It includes following steps:

1) Download the contents/article to be extracted.
2) Extract the article from the html.
3)Figure out the 3 or 5 most important sentences from the article.

A. Algorithm of existing system

1) Download the Article from URL.
2) Get rid of html tags and everything else other than the article (use beautiful soup).
3) Split the article into words. (Use NLTK function like word-tokenize and sent-tokenize).
4) Eliminate the stop words. (Is, this, the, a)
5) Find how often each remaining word is repeated. (Frequency of particular word in the article).
6) The more common the word appears, the more important it is. So, for each sentence, find a score of how important the words in the sentence are.
7) Rank the sentence by that score. (Select top 3-5 sentences)

B. Pros of existing system

They are quite simple since they don't make changes in the document the just try arranging them in the form of highest priority. They use existing natural language phrases which are in the input of the model.

C. Cons of existing system

They miss flexibility since there is no use of grammar, figure of speech and they also lack use of novel words or connectors. It's impossible for them to explain or summarize like people do.

IV. PROPOSED SYSTEM

We are going to take news articles from the news website's such as inshorts, news hunt, etc. And then we are going to scrape the website to get the new article for this we have created tool with the help of python language. This tool basically does is it extract data from website which is HTML file. The reason to do this is collecting the data in order to give instant news to the user. For scrapping the article from different websites, we will be using python and it's libraries like urllib, beautiful soup, etc. The python script will be run on bash terminal in order to get real time articles from the website and the extracted output. With the help of bash terminal, we will host our flask application. Flask is python-based web framework. Extractive text summarization work by picking a selective portion of the input text and then condensing the text to summarize, unlike abstractive techniques which conceptualize a summary and paraphrases it. For pickling we use pickle library it is a native python library to save (serialize) and load (de-serialize) python object as files on the disk.

Steps:

1. Train the model using Jupiter notebook.
2. Save the trained model object as a pickle file (serialization).
3. Create a flask environment that will have an API endpoint which would encapsulate our trained model and enable it to receive inputs (features) through GET requests over HTTP/HTTPS and then return the output after de-serializing the earlier serialized model.
4. Run the flask script along with the trained model on bash terminal.
5. Make requests to the hosted flask script through a website.

We use Extractive algorithms like Lex Rank and LSA.

1.LexRank is an unsupervised graph-based approach similar to Text Rank. IDF-modified used by Lex Rank Cosine as the similarity measure between two sentences. This similarity between two sentences is used as weight of the graph edge. Lex Rank uses a technique which makes sure if the all the sentences with high priority are not similar to each other. Lex Rank is a new method of determining the most important sentences in a given corpus. The problem of extracting a sentence that represents the contents of a given document or a collection of documents is known as extractive summarization problem. In extractive summarization problem, we want to extract one representative sentence that capture as broad as possible the content of the corpus,

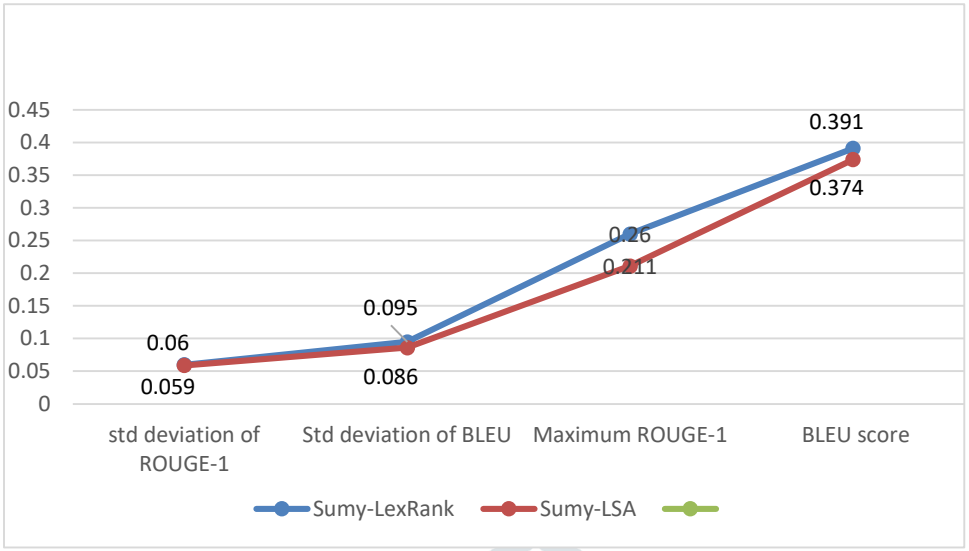
whether it is one document (single document summarization) or several documents (multi-document summarization). The new method, named Lex Rank, is identified by PageRank method. This method works firstly by generating a graph, composed of all sentences in the corpus. Every sentence represents one node, and the edges are similarity relationship between sentences in the corpus. In this research, they measure similarity between sentences by considering every sentence as bag-of-words model. This means that the similarity measure by frequency of word occurrence in a sentence. Frequency contributes to the similarity strength as the number of word occurrences is higher. At the same time, the inverse document frequency regards low frequency words inversely contributes to higher value to the measurement. This is then used as a measurement for similarity between sentences basically, calculating the 'distance' between two sentences x and y. More the similar two sentences, more the 'closer' they are to each other. Thus, the similarity measure is used to build a similarity matrix, which than can be used as similarity graph between the sentences. The Lex Rank algorithm calculates the importance of sentences in the graph by considering its relative importance to its neighboring sentences, where a positive contribution will raise the importance of a sentence's neighbor, while a negative will lower the importance value of a sentence's neighbor. This idea is primary the same with PageRank, unless it is used in counting the importance of sentence in a given set of sentences. To extract the most important sentences, from the resulting similarity matrix we apply a threshold
ding mechanism. A threshold value is used to filter out the relationships between sentences whose weights are fall below the threshold. The result is a subset of the similarity graph, from where we can pick one node that has the highest number of degrees. This node is considered salient or represents a summary sentence of the corpus.

2. Latent semantic analysis is an unsupervised summarization method which along with finding the frequency of important terms it decomposes to find the singular value for better and efficient summarization. It is one of the most recent suggested technique for summarization. LSA works by projecting the data into a lower dimensional space without any heavy loss of information. Latent semantic analysis uses spatial decomposition. In spatial decomposition the singular vectors of the words which recurring in the corpus. The higher the magnitude of the singular value the higher is the importance of the word in the document. Latent Semantic Analysis (LSA), also known as Latent Semantic Indexing (LSI) literally analyse the documents to find underlying meaning or concepts of those documents. If each word only meant one concept, and each concept was only described by one word, then LSA would be easy since there is a simple mapping from words to concepts.

Unfortunately, English has different words and ways to represent the sentences which have similar meanings. There are many words with same meaning(synonyms), words with multiple meanings, and all the other type of ambiguities and redundancy in meaning. Which have hard time understanding.

V. RESULTS

We used two extractive algorithm Lex Rank and LSA. The output of both the algorithm was too good then the existing system. To evaluate the text summarization quality, we used ROGUE-N metric and BLEU metric. Rouge-N is a word N-gram is a measure of how much efficiency between the model and the summarized output. It finds the ratio of the no of counts of phrases which occur in both model and summary known as N-gram. BLEU metric is a modified form of precision, extensively used in machine translation evaluation. Precision is the ratio of the number of words that co-occur in both gold and model translation/summary to the number of words in the model summary. Unlike ROUGE, BLEU directly accounts for variable length phrases – unigrams, bigrams, trigrams etc., by taking a weighted average.



For extractive techniques, our graph tells us that Lex Rank outperforms LSA. We believe that the dataset which is used affects the quality of obtained summaries. A good practice would be to run both the algorithms and use the one which gives more satisfactory summaries.

VI. CONCLUSION AND FUTURE SCOPE

We are able to summarize news articles by analyzing the content of the news. In this process we analyze that each and every website uses different pattern to display their news so it is difficult to scrap data from various sites. Hence now we are able to scrap data from inshort news website which allow us to scrap article and body from same page. After we used two extractive algorithms i.e. Lex Rank and LSA because the results were too good then existing system. We intend to create a summarization model which can create a summary of news articles which are generated every day. This will help the users to get whole idea of the news without reading the entire news.

In future we are going to use news.API which allows us to get news, headlines, articles from over 30,000 news sources which allows us to import various type news in our website and more over we are going to build mobile application for both Android and IOS platform which becomes handy to people to read news anywhere anytime. And if we get access to enough resources then we can switch to abstractive method which can allows news to summarize in more contextual form making the summary more precise and correct. We may also build a complete automatic process pipeline for fetching news, scraping the news and then summarizing and displaying it.

VII. ACKNOWLEDGMENT

Our grateful thanks to Dr. Ashok Kanthe (Head of Department of Computer Engineering) who in spite of extraordinary busy with his duties, took time out to hear, guide and keep us on the correct path. We choose this moment to acknowledge his contribution gratefully. I also indebted to Dr. Ashok Kanthe, for extending the help to academic literature.

We express our gratitude to Dr. Madhumita Chatterjee (Principal) for their constant encouragement, Co-operation and support.

REFERENCES

[1] M. Haque, et al.," Literature Review of Automatic Multiple Documents Text Summarization", International Journal of Innovation and Applied Studies, vol. 3, pp. 121-129, 2013.

[2] N.Moratanch ,S.Chitrakala "A Survey on Extractive Text Summarization" IEEE International Conference on Computer, Communication, and Signal Processing (ICCCSP-2017)

[3] Gune ¨ ,s Erkan, Dragomir R. Radev, "LexRank: Graph-based Lexical Centrality as Salience in Text Summarization". Department of EECS University of Michigan, Ann Arbor, MI 48109 USA Journal of Artificial Intelligence Research 22 (2004) 457-479.

[4] Haroran Li, Junnan Zhu, Cong Ma, Jiajun Zhang and Chengqing Zong, "Read, Watch, Listen and summarize: Multi-model Summarization for Asynchronous Text, Image, Audio and Video", IEEE Transaction on Knowledge and data engineering, Vol. X, No. Y, Month year.

[5] Prakhar Sethi, Sameer Sonawane, Saumitra Khanwalker, R.B. Keskar "Automatic Text Summarization of News Articles" 2017 International Conference on Big Data, IoT and Data Science.

[6] Luciano Cabral, Rinaldo Lima, Rafel Lins".2015 Fourteenth Mexican International Conference on Artificial Intelligence, "Automatic Summarization of News Articles for Mobile Devices" 978-1-5090-0323-5/15 2015 IEEE.

[7] R. Mihalcea, "Graph-base ranking algorithms for sentence extraction, applied to text summarization", in Proceedings of ACL 2004 on Interactive poster and demonstration sessions, 2004.

[8] Rajnish M. Rakholia & Dr. Jatinderkumar R. Saini "Lexical Classes Based StopWords Categorization" 978-1-5090-3480-2/16/$31.00 2016 IEEE.

[9] P.Krishnaveni & Dr.S.R. Balasundaram "Automatic Text Summarization by Local Scoring and Ranking for Improving Coherence" 978-1-5090-4890-8/17/$31.00 2017 IEEE.