Text Summarization using Extractive and Abstractive Methods

Saurabh Varade[1,*], *Ejaaz* Sayyed[1,**], *Vaibhavi* Nagtode[1,***], and *Dr. Shilpa* Shinde[1,****]

[1]Department of Computer Engineering, Ramrao Adik Institute of Technology, India

Abstract. Text Summarization is a process where a huge text file is converted into summarized version which will preserve the original meaning and context. The main aim of any text summarization is to provide a accurate and precise summary. One approach is to use a sentence ranking algorithm. This comes under extractive summarization. Here, a graph based ranking algorithm is used to rank the sentences in the text and then top k- scored sentences are included in the summary. The most widely used algorithm to decide the importance of any vertex in a graph based on the information retrieved from the graph is Graph Based Ranking Algorithm. TextRank is one of the most efficient ranking algorithms which is used for Web link analysis that is for measuring the importance of website pages. Another approach is abstractive summarization where a LSTM encoder decoder model is used along with attention mechanism which focuses on some important words from the input. Encoder encodes the input sequence and decoder along with attention mechanism gives the summary as the output.

1 Introduction

Internet contains vast amount of data. World Wide Web provides huge variety of data from which the user can find useful data according to their needs and purpose. So, it is highly beneficial to use an efficient text summarization in this hectic world so that only user required and useful data is provided to the user in a lesser time.

Manual summarization of documents and web-contents is too time-consuming this paved the need for automatic summarization [8, 10] of text documents and web-contents. Summary created with the help of main points which are the sentences form the original document with the help of a software is known as automatic summarization.

Text Summarization can be classified into different categories:

- Single document and multiple document
- Generic and query-based approach
- Extractive and Abstractive

Text summarization can be applied to single document as well as it can be extended to multiple documents. The summarized version of the original set of documents should be context based. LexRank like ranking algorithm can be used for multiple document summarizations. TextRank algorithm can be used for single document summarization.

Text summarization can be classified based on the purpose, input, and method that are opted. Automatic summarization has two types of approaches which are extractive and abstractive summarization. Selecting some word, phrases or sentences form the original text to form a summary is known as extractive summarization [11]. Abstractive summarization works by building a semantic similarity representation and using the techniques of natural language generation to generate a summary which is similar to manual summary.

A reader and an identifier are required to select between prime and unnecessary words/sentences in the text document to generate a summary of a large text document. Clustering similar document and presenting a summary is also provided by the document summarization. A good summary generator should be capable of reflecting the different fragments from the document while keeping the redundancy level low.

In today's world where there is an over-abundance of data and lack of manpower with less time to interpret the data, an automatic method for text summarization is necessary. A system is proposed named Data Summarization using Unsupervised Learning which will summarize the data from these documents or article and will provide a summary for that data of random length.

The main objective of the project is to implement a system which summarizes the data in a proper manner preserving its overall meaning and contents also without changing its context. Aim is also to reduce the time and provide a better result for the end user.

The organization of report is as follows. Section 2 will provide an introductory and brief review of related works along with some important facts. Section 3 will elaborate details of the system (Data Summarization Using Unsupervised Learning) and its implementation. Section 4

*e-mail: saurabhvarade123@gmail.com
**e-mail: ejaaz.sayyed.common@gmail.com
***e-mail: vaibhavinagtode1@gmail.com
****e-mail: shilpa.shinde@rait.ac.in

will present experimental results obtained and the analysis work. Section 5 will provide the conclusion for our work.

2 RELATED WORK

A survey of the research done for data summarization and the currently existing systems, give the following results.

A. **Automatic text summarization using semi-supervised learning [1]**

Idea mentioned in this paper includes development of summarization which is fully automated. This is the main idea and it uses unsupervised and semi-supervised learning. Linear classifiers which are simple have been used for implementation. Increase in performance can be clearly seen in experiments on Reuter's news-wire.

Half of the performance increase, which is allowed by a fully supervised system can be reached by unsupervised learning. This is realistic for related applications. The way how query based summaries are used is helpful.

B. **Review Paper on Extractive Text Summarization [2]**

It gives an over-all idea of Extractive Summarization with its features. A summarization machine is given which provides an idea about how the summarization process works. Query driven and generic are the two basic types update of the summary.

The extractive text summarization methods are preferred depending on TF-IDF (Term Frequency-Inverse Document Frequency), cluster based methods, the single value decomposition is preferred by the Latent Semantic Analysis (LSA) and the vector space model is preferred by the concept based summarization.

Advantages:

- Precise information can be helpful to understand the document more effectively and efficiently.
- Summarization can make searching easier.

Limitations:

- Extraction of crucial information or optimization of the whole document in order to minimize the time required to review the document.
- Generation of summary with minimum redundancy, maximum relevancy and referring elements of document in the summary.

C. **Extractive Text Summarization Using Graph Based Ranking Algorithm And Mean Shift Clustering [3]**

An introduction to Abstractive and Extractive summarization is given along with Extractive summarization types. A brief introduction to Graph based ranking algorithm is mentioned. Concepts used are Text Rank (Single document), Lex Rank (Multiple documents), Co-Rank, Mean-shift Clustering and ROUGE Evaluation.

Advantages:

- Mean Shift Algorithm is general, application-independent tool.
- Model-free, doesn't assume any prior shape on data clusters.

Limitations:

- Output depends on window size.
- Computationally expensive.

D. **Summarization Using Clustering and Classification: Spectral Clustering Combined with k-Means Using NFPH [4]**

In classical k-means, the cluster centroid's initialization method can be replaced by this system. This can solve some of the limitation which occur in k-means algorithm.

Advantages:

- From the output it is cleared that the accuracy was improved by 2

Limitations:

- The processing time has increased from 1 to 6 seconds . This is due to fact that an additional step is required for performing the NFPH based initialization of k-means.

E. **Enhancing unsupervised neural networks based text summarization with word embedding and ensemble learning [5]**

Bag of words method and sentence to vector representation are the two implementation models used. Each word is represented as vector in word2Vec. Similarly, in sentence2Vec, each sentence in the text document is represented as vector in an embedded low-dimensional space.

Two summarization frameworks are developed based on unsupervised deep learning. This is called as Auto-Encoder and Variation Auto-Encoder. Computer vision and bioinformatics has been successfully applied with extreme learning machine which has become the state of the art learning framework.

Promising results have been gained in English and Arabic languages by models applied in different languages. Graph model is the approach used for the summarization for the Arabic newspaper.

Advantages:

- Word2Vec approach (already published by Google) improves the results obtained by unsupervised deep learning models.
- Data fusion(voting) improves the quality of the output summary as the fusion is of the BOW and Sentence2Vec transformation which further re-ranks sentences.

Limitations:

- Needs big corpus with a large number of text documents in order to build powerful models with a more discriminative feature space.
- It is very hard to find the optimal parameters for the adopted neural network models.
- Learning process is time-consuming because the models are trained on a large number of documents.

F. **Unsupervised text summarization using sentence embeddings [6]**

There are many applications of Dense vector representation in NLP. There was text summarization in the past which used the clustering of sentences in a high-dimensional space. But, those systems used TF-IDF instead of sentence embeddings.

Latent semantic indexing to identify sentences that explain the latent concepts very nicely in the document were used by another class of vector based methods. Work of Skip thought vector space has the main goal of encoding sentences in vector using RNN with LSTM.

Unsupervised text summarization approach is proposed by clustering the sentence embeddings trained to embed paraphrases near each other. A summary is generated from the cluster of sentences by selecting a representative from each cluster.

The text whose embedding is the nearest according to Euclidian distance that is, to the centroid of the cluster, are chosen by the extractive method.

The embeddings are decoded into sentences by the decoder in abstractive method. This is done by using the Recurrent Neural Network (RNN) with Long Short Term Memory (LSTM). Skip thought embeddings have less precision than the system which uses Paragram embeddings.

Advantages:

- Systems using Paragram embeddings have a higher precision than those using Skip-thought embeddings.
- K-Means proved to be better than Mean shift clustering performance wise on the tested datasets.

Limitations:

- The decoder tended to generate a fair number of <UNK> tokens, which is possibly because the tested dataset is a scientific papers, which has a number of words that do not occur frequently enough in the dataset itself, leading to poor parameter estimates in the decoder.
- No clear trend between the different types of abstractive systems that is consistent across clustering methods.

G. **A survey on LSTM memristive neural network architectures and application [7]**

Long short-term memory (LSTM) consists of state memory and multilayer cell structure which is a special type of recurrent neural network (RNN). The paper has explored LSTM architecture and different variants of lstm architecture. Various applications such as prediction, pattern classification, analysis, etc are mentioned with lstm models.

3 METHODOLOGY

3.1 Proposed Work

3.1.1 Extractive Summarization

The proposed system uses vectorization of sentences, TextRank algorithm, cosine distance calculation.

The proposed system comprises of three main modules:

- First the text preprocessing, cleaning and similarity construct.
- Second the Graph based ranking (in this case) with appropriate ranking algorithm and techniques.
- Lastly, the selection of the k highly ranked sentences as the summary of the text input.

3.1.2 Abstractive Summarization

The proposed system uses a LSTM [9] encoder decoder model along with attention mechanism.

The Proposed system comprises of main modules:

- First the text preprocessing and tokenization
- Second comes an encoder model which encodes the input text and pass it to the decoder.
- Third, the decoder and the attention mechanism produce the output that is the summary of the input text by selecting the representation from the final states.

3.2 Techniques

3.2.1 Extractive Summarization

The preprocessing and cleaning step consist of removing the stop words. Stops words are the words which are commonly used in a language. Removal of stop words helps in the model to focus on ranking based on the important words or words with significant usage thus can be beneficial in reducing the variance. Then, the sentences are vectorized. The previous proposed systems use Bag of Words (BOW), which is extended in this model to Vector of sentences, so as to achieve the extractive nature of the model.

The similarity construction of the sentences is done with cosine distance calculation. Similarity between two vectors of an inner product space is measured by cosine similarity. The vectors are plotted on a two-dimensional plane and the cosine of the angle between the vector gives how close or similar these vectors could be. Similarity will be one if cosine of the angle is calculated between two same vectors.

3.2.2 Abstractive Summarization

The preprocessing and cleaning step consist of removing the stop words, URL's, html tags, digits, punctuations, contractions. Doing this helps in the model to focus on contextual information. This model is using Many-to-Many Seq2Seq model which contain an encoder and a decoder.

Here, long term dependencies are captured using LSTM (Long Short Term Memory) as an encoder and decoder components. This helps on overcoming the problem of vanishing gradient. Converting the input sequence into a vector is done by encoder and the output sequence is predicted by the decoder with the help of attention mechanism.

Attention mechanism helps the decoder by focusing more on some words in the input sequence.

3.3 Design of the System

Figure 1 shows the flow of extractive summarization and Figure 2 shows the flow of abstractive summarization.

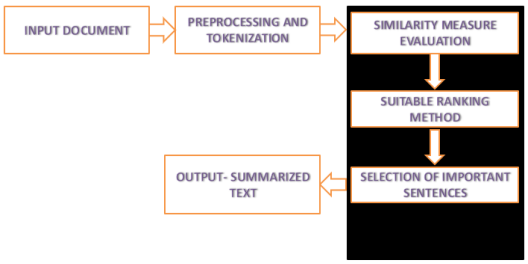

Figure 1. Flow of Extractive summary generation

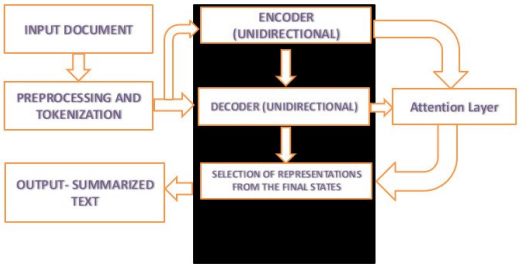

Figure 2. Flow of Abstractive summary generation

4 RESULT AND ANALYSIS

The techniques used in the extractive summarization, that is, the cosine distance and the TextRank algorithm(in NLP, also called text Rank algorithm) makes the system computationally efficient and can be used for many NLP based applications. The time complexity of the implemented extractive summarization is linear, that is it requires time of O(n). The system can randomly generate any number of summary sentences, depending on the size of the text given as input. The accuracy of the abstractive summarizer model tends to be around 60% for unseen data. For the extractive summarization, the accuracy varies as the sentences in the summary may or may not be much reliable.

The loss incurred for the validation or test data in the training phase is decreasing with a plateau like line. Soon after 18 epochs, the model starts to learn noise data in the training data, indicating possibility of overfitting. The model then breaks the training phase and returns best parameter values of the model to have been achieved. Figure 3 shows the graph depicting the loss of the model during the training phase for abstractive summarization. The X-Axis shows the number of epochs and the Y-Axis shows the loss function value during the training of the model. Figure 4 shows the graph depicting the accuracy of the model during the training phase for abstractive summarization. The X-Axis shows the number of epochs and the Y-Axis shows the Accuracy metric during the training of the model.

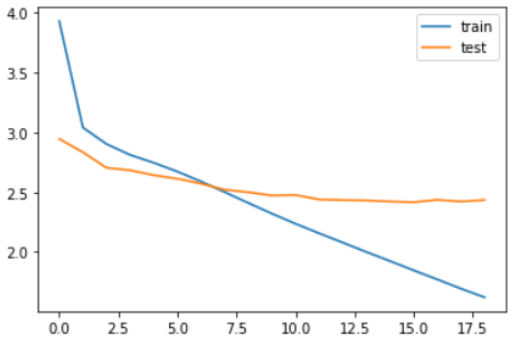

Figure 3. Loss graph for abstractive summarization

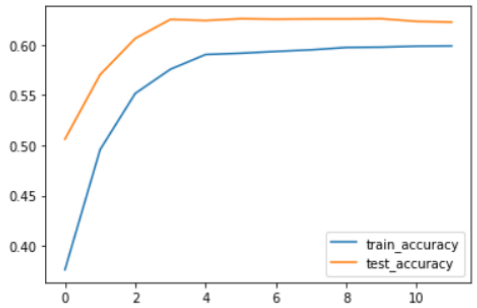

Figure 4. Accuracy graph for abstractive summarization

5 CONCLUSION AND FUTURE WORKS

Comparing the two-systems performance-wise, the result of extractive summarization will be more efficient and but less accurate than abstractive summarization, as the dependency of the abstractive summarizer will be more on its model and hence might take a significant amount of time to process a small amount of data. The accuracy of

the abstractive summarizer model tends to be around 60% for unseen data. For the extractive summarization, the accuracy varies as the sentences in the summary may or may not be much reliable.

In future, implementation of the current system in different domains like android application, standalone software or integration with multi-tools applications can be done. A working system for different research papers that can handle all type of notions from different streams like medical, astronomy, etc. can also be implemented. Further, more fine tuning and improving the abstractive summarizer for long documents and improving the performance for small documents for better understanding and intuition of working of the model can be achieved.

6 ACKNOWLEDGMENT

The authors thank the many people who have done lots of help for us and provided useful guidance for our paper.

References

[1] A. Zamanifar, B. Minaei-Bidgoli, M. Sharifi, " Automatic text summarization using semi-supervised learning", ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 635 (2008)

[2] A. Sahoo, Dr.A. Kumar Nayak, " Review Paper on Extractive Text Summarization", IJERCSE, **5**, (2018)

[3] Ramesh, Reema, Rajan, Binu"Extractive Text Summarization Using Graph Based Ranking Algorithm And Mean Shift Clustering", ICRTCCNT, (2019)

[4] N. Sapkota, A. Alsadoon, P. W. C. Prasad, A. Elchouemi, A. K. Singh, " Data Summarization Using Clustering and Classification: Spectral Clustering Combined with k-Means Using NFPH", COMITCon, 146-151 (2019)

[5] N.Alami, M. Meknassi, N. En-nahnahi, " Enhancing unsupervised neural networks based text summarization with word embedding and ensemble learning", Expert Syst. Appl., **123**, 195-211 (2019)

[6] Padmakumar, Aishwarya, A. Saran, " Unsupervised Text Summarization Using Sentence Embeddings", (2016)

[7] Smagulova, Kamilya, James, Alex, " A survey on LSTM memristive neural network architectures and applications", The European Physical Journal Special Topics, **228**, 2313-2324 (2019)

[8] "Introduction to summarization in machine learning", https://towardsdatascience.com/a-quick-introduction-to-text-summarization-in-machine-learning-3d27ccf18a9f, Accessed: 2020-12-02

[9] "Understanding LSTMs", https://colah.github.io/posts/2015-08-Understanding-LSTMs/, Accessed: 2021-01-15

[10] "Automatic Summarization", https://en.m.wikipedia.org/wiki/Automatic_summarization, Accessed: 2020-10-06

[11] "Introduction to textrank in python", https://www.analyticsvidhya.com/blog/2018/11/introduction-text-summarization-textrank-python/, Accessed: 2020-09-06