Bengali Abstractive Text Summarization Using Deep Learning

BY

Md. Ashraful Islam Talukder
ID: 161-15-7100

Abu Kaisar Mohammad Masum
ID: 161-15-6759

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Sheikh Abujar
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised By

Md. Sadekur Rahman
Assistant Professor
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

DECEMBER 2019

APPROVAL

This Project titled **"Bengali Abstractive Text Summarization Using Deep Learning"**, submitted by Md. Ashraful Islam Talukder, ID No: 161-15-7100 & Abu Kaisar Mohammad Masum, ID No: 161-15-6759 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 6th December,2019.

BOARD OF EXAMINERS

Dr. Syed Akhter Hossain — **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Nazmun Nessa Moon — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Fizar Ahmed — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Mohammad Shorif Uddin — **External Examiner**
Professor
Department of Computer Science and Engineering
Jahangirnagar University

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Sheikh Abujar, Lecturer, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

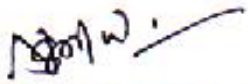
Supervised by:

Sheikh Abujar
Senior Lecturer
Department of CSE
Daffodil International University

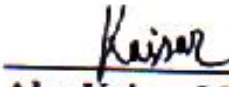
Co-Supervised by:

Md. Sadekur Rahman
Assistant Professor
Department of CSE
Daffodil International University

Submitted by:



Md. Ashraful Islam Talukder
ID: 161-15-7100
Department of CSE
Daffodil International University



Abu Kaisar Mohammad Masum
ID: 161-15-6759
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

At first, we explicit gratitude to almighty for his perfect gift to makes us conceivable to finish final year project phase successfully.

We would like to thanks our supervisor **Sheikh Abujar** for his proper advice to complete this excellent research work for the Bengali language. His support and instruction provided us with the courage to complete this research project accurately. He served us all related resources and relevant information to do this research for the Bengali language. We also thank our co-supervisor **Md. Sadekur Rahman** supports us to complete this work. We are truly a gratitude to our department head, **Dr. Syed Akhter Hossain** for his valuable support to do such kinds of research work in the Bengali Language. Also, like to thank other faculty members and the staff of our department for their supports.

We are very thankful to our DIU NLP and Machine Learning Research Lab to provide us with instruments and gives facility to complete this research work.

Finally, we thank and gratitude to our family and friends for their support and the inspiration us to finish this research work successfully.

ABSTRACT

A human can describe his mood with the help of text. Therefore understanding the meaning of the text is very important. Sometimes, it is hard to understand the meaning of those texts, alongside this is also time consuming. The machine is the best way to solve this problem. As a part of machine learning, text summarization is a large field of research in natural language processing. Build automatic text summarizer is the main focusing point of all research. Text summarizer produces the gist part of a large document in a short time. Automatic text summarizer for other languages has made previously but not for the Bengali language. Increasing the tools and technology of Bengali language is the main goal of this research. In this research work, we've tried to build an automatic text summarizer for the Bengali language. Though, working with the Bengali language was a very challenging part of this research. But until the end, we have made a base for automatic text summarizer for the Bengali language. The dataset used is collected from online social media. The deep learning model is used to make the summarizer. In the model, train time reduce the loss is directly affect the experiment result. We have reduced the training loss of our summarization model for Bengali text summarizer and which are capable to generate a short text summary.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATION

NMT	Neural Machine Translation
NLP	Natural Language Processing
GNMT	Google's Neural Machine Translation
RNN	Recurrent Neural Network
DRGD	Deep Recurrent Generative Decoder
CNN	Convolutional Neural Network
NLTK	Natural Language Tools Kit
RNN	Recurrent Neural Network
LSTM	Long Short Term Memory

CHAPTER 1

Introduction

1.1 Introduction

In the field of text summarization there are two categories. Abstractive and Extractive text summarization. Abstractive text summarization contains an abstract of the text document. Basically providing abstract is the representation of the main idea of the text but here summarizer does not repeat the original sentences. Here is the main challenge to finding the gist of the text in natural language processing. The maximum number of research work are held on the extractive text summarization. Extract keyword and find the most frequent words from the text is the main idea of extractive text summarization. But generate a new word or sentences based on text is the most challenging stuff. This is not mandatory to have the word in the providing abstractive summary is also present in the original context. There are many abstractive text summarization research work has done previously in different languages. In this time, we have tried to build an abstractive text summarizer for Bengali language applying deep learning algorithms.

Bengali is one of the most usable languages over the world. Increasing the tools and technology for this language is very important. Therefore, the research area of Bengali language needs expansion. An automatic system text needs to be processed. NLP tools and library very much helps to process any kinds of text. Working in the Bengali language to build an automatic system is difficult compared to other languages. Because some NLP libraries are not built for the Bengali language therefore, all techniques and libraries are applying by raw coding. Our research work can provide abstractive text summary for Bengali text. No machine gives 100% accurate results every time but maximum time a satisfactory result can be obtained. Our automatic abstractive text summarizer is also looked like that. All the generated summaries are not 100% accurate but the maximum response of machine summary is satisfactory for Bengali text summarization.

1.2 Motivation

Text summarization is a faster way to summarize a long text or text document. Extract the main keywords and make a relevant gist of corresponding text is the main idea of summarization. Reading long text fluently and also find the main abstract form text is very painful and time consuming. Sometimes when we read a long text but can't understand the meaning. If the document size is multiple documents, it is much more difficult to find the abstract. Therefore, automatic text summarizer is a tool that can help to summarize the text within a short time. Automatic text summarizer also identifies the total documents, words, total frequented word and which words are important for the text document.

Today we spend a lot of time in social media, reading web pages, news articles and blogs but after some time may fill bored. Cause of unstructured data and unclear meaning. That is also a cause which means to need a text summarizer. Abstractive text summarizer is text summary approach which find the gist part from the given text documents, that's not mandatory if containing summary's words present or not in the original text documents.

Data is one of the most valuable things in this modern world. Every day a huge number of text data produced from different sources. This huge number of data need huge memory space which is most costly and creates a problem in putting spaces. Therefore, summarizer makes a summary of those long text and reduce the size of the document and put only core information by removing the unnecessary text. That's why an automatic text summarizer is required for the modern technology.

Bengali is our mother language. NLP resource for this language is not sufficient for the user. That's why we need to build NLP resources and tools and technologies. So the main focus of this research is building an automatic abstractive text summarizer of the Bengali language to reach the Bengali NLP treasure.

1.3 Rational of the study

History of the Bengali language is very rich. Today millions of people use the Bengali language as their native language. But in this modern era, the tools and technology of Bengali language are not rich like other languages. Therefore, we need to increase the technologies for this language. Most of the text related problems can be solved by the NLP tools and techniques. Text summarization is a core problem of NLP. A text summarizer contains the abstract of a long sequence of text. It helps human to understand the meaning of a long text easily with a fluent and error-free summary. Most and major NLP techniques already builds for other languages such as English, French, Chines etc. But for Bengali text a few models have been build which is not enough. Therefore, the research area of Bengali NLP needs to be increased. The main obstacle for Bengali text is preprocessing. The machine cannot understand some of the Bengali characters and symbols. To handle this problem needs to use the Unicode of those characters or symbol. NLTK library is not available for Bengali text. That's why Bengali tools do not perform accurately as like as other languages. Research is the only way to provide a solution to these kinds of problems. So, in this research work, we try to show how to processes Bengali language and make abstractive text summarizer for the Bengali language. That helps us to reduce the size of the document and provide a fluent summary in shot time.

1.4 Research Questions

- What is Bengali text summarization?
- How Bengali text summarization works?
- What are the benefits of Bengali text summarization?
- What are the differences between Bengali and English text summarization?
- How to preprocess Bengali text in NLP?
- What are the future works of Bengali text summarization?
- How Bengali text summarization Model works?

1.5 Expected Output

Since this is a research project, our main concern was to publish research paper in a related field. Research work always a continuous process. Many people analysis specific research topics to find an efficient solution. Then the developer develops the tools for the users. The maximum number of research work and tools are developed using extractive text summarization in the Bengali language but not in abstractive text summarization. Also, many researcher and developer are not interested to provide their dataset and resources for everyone. As a result, some research work is no longer been used. In Bengali language text summarization is a new research topic. Some research works are done in previous for Bengali text summarization. But the result was not enough satisfactory for making an automatic Bengali text summarization. An automatic system is dependent on the machine. Therefore the machine needs to learn. Then the learning model is working in the backend of a system such as a web or mobile application. In this research, we introduce a machine learning method for abstractive Bengali text summarization and show to necessary steps on how to build a model for automatic Bengali text summarization.

1.6 Report Layout

In this report have a total of 5 chapters. Chapter 1 contains an overview of the whole research work. It has some sections such as 1.1 Introductions of the work, 1.2 Motivation of this research, 1.3 Rational Study of the search, 1.4 Research Questions, 1.5 Expected Output and 1.6 Reports Layout of the research. In Chapter 2 we have discussed about Background Studies of the research and its subsections are 2.1 Introductions, 2.2 Related works, 2.3 Research Summary, 2.4 Scope of the Problem, 2.5 Challenges. In Chapter 3 we have discussed the whole Research Methodology with subsections 3.1 Introduction, 3.2 Research Subject and Instrumentation, 3.3 Data collection procedure, 3.4 Statistical Analysis of Datasets, 3.5 Implementation Requirements. In Chapter 4 Experiment and Results of the research are discussed and the subsection is 4.1 Introduction, 4.2 Experimental Results, 4.3 Descriptive Analysis, 4.4 Summary. Chapter 5 contains the Conclusion and future works of the research with the subsections 5.1 Summary of the Study, 5.2 Conclusion, 5.4 Implication for Further Study. End of all section given the references which helped us in our research work.

CHAPTER 2

Background Studies

2.1 Introduction

Text summarization is a process which makes a short view of long text or long text document. Text document has a long sequence of text. Finding short, eloquent and understandable summary is the main spotlight of text summarization. Making a summary of a long text document for a human is time-consuming and costly. Therefor automatic text summarization is a great way to help a human. Machine learning approaches are the only way to develop an automatic system. Automatic text summarizer helps us to reduce the size of the document and save memory space which also reduces the cost of space.

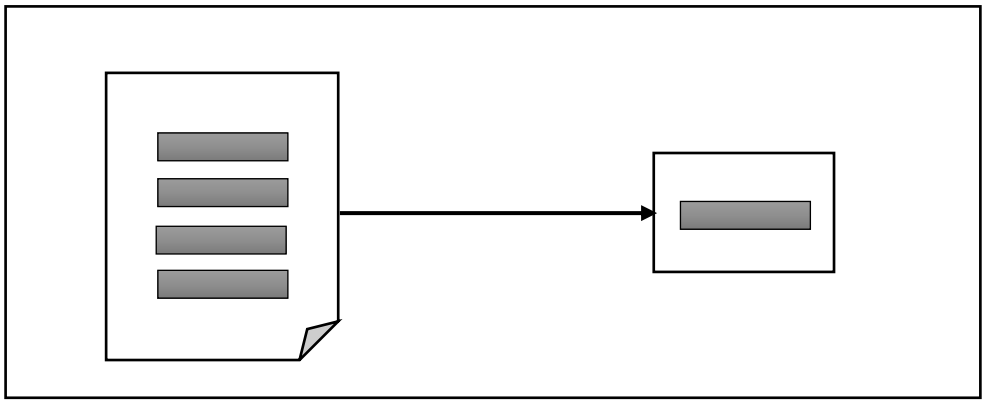


Figure 2.1.1: Text summarization

Every day a huge number of document processes on the internet such as Facebook posts. A large number of users finding or search information form a document in time. But saving and find the information from those document is a complex process and also need a huge space to store those amount of information. Therefore, the summarization technique is used to solve this problem with an efficient, fast and effective way. In memory, save the abstract, main words and sentences of text document then required they serve the information very quickly. There are two types of text summarization: Abstractive and Extractive text summarization. Abstractive text summarization contains abstract of the long text which is not mandatory to present or not present the text document. Where extractive text summarization contains the main words, phases of the original text document to summarize text document.

2.2 Related Work

Text summarization is the most researchable topic in natural language processing. Many research work has been done in this field for different language. Major research was held on extractive text summarization but few in abstractive text summarization. This section will be discussed about some noble works in these fields.

Any language modelling is dependent on machine translation. Machine Translation helps a machine to understand the processed text and helps to make an automatic system. NMT is a new approach to machine translation. Bahdanau et al [1] introduce a new approach for NMT. They used joint learning to align and improve the performance of normal encoder and decoder methods. Vector of the text sentences is worked input the encoder where the decoder generates the possible output of the vector sequences. After that increasing the effectivity of NMT, the Attention-Based [2] methods are introduced in later. NMT has some limitations such as training and testing are more expensive and can't provide a good result for a rare word in the sequence. Wu Y et al [3] presented a GNMT system for reducing the complexity of NMT.

RNN is a technique to solve the text related problems. Nallapati et al [4] show sequence to sequence learning using RNN's give outperforming result for abstractive text summarization. Encoder contains the fixed length of the input sequence of the vector and decoder generates the most related sequence of the encoded sequences. Improve the performance of abstractive text summary DRGD is used [5] in recent time. Reinforcement learning is the modern use in abstractive text summarization. Wang et al [6] proposed a model using reinforcement technique for abstractive text summarization. This time they use convolutional sequence learning for abstractive text summarization. Contextual learning of text is the main idea of this work. The whole process is dependent the CNN [7].

Ilya Sutskever et al [10] introduce a method for seq2seq learning using multilayer LSTM. One maps the input sequence form the target text vector which is encoder. Another one decodes the sequence vector which is the decoder. Therefore, the LSTM has not any complexity in working with the long sequence. Reversing the order of sequence is possible in this methodology.

For the summarization of long term text such as Wikipedia extractive text summarization used widely. Peter J. Liu et al [11] introduce a decoder instead of encoder and decoder model which is

able to generate fluent summary from long text sequence. Also this method is able to generate multi-document summary from a large and paralleled dataset.

Lifeng Shang et al [12] give a methodology for short text summarization. Decoding processes of the text contain the summary of the long text sequence. Normal encoder and decoder give a good response for short text summarization. Probabilistic calculation of the text is a help to generate the summary of the text.

In this research work we have used sequence to sequence learning for making abstractive text summarization. Follow the attention mechanism for making the abstractive text summarization. Using dataset contains the short Bengali text and their corresponding summary. After the using of encoding and decoding mechanism model provide a good result for the Bengali abstractive text summarization.

2.3 Research Summary

In our research, we have introduced a methodology for Bengali abstractive text summarization. We build a model using deep learning. To build this model we have used our own dataset. Dataset is collected form social media. At first collect Bengali status, comment, page and group posts from Facebook. Then create a summary of each Bengali text. Therefore, the dataset contains two columns, one is Bengali text and another is their corresponding summary. The total number of two hundred data with their summary in the dataset. Before creating a deep learning model we have preprocessed the Bengali text. In the preprocessing stage, at first split the text and then add Bengali contractions and remove stop words from the text. After preprocessing we have count the vocabulary of whole data. Word embedding is important for deep learning model. Word vector helps to save the related vocabulary in a file with a numeric value. We used a pre-trained word vector file for Bengali text which is available in online. We build a sequence to sequence model based on attention model. In this model encoder and decoder is used with Bi-directional LSTM cell. Word vector is the input of the encoder and relevant word vector in the decoder is the output of the model. An encoder and decoder to pass the sequence need a token which is known as a special token such as PAD, UNK, EOS etc. After declaring and define all function and library we train the model for more than 3 hours. Then we found a good response from the machine.

2.4 Scope of the problem

NMT is a new approach to machine translation. It depends on the encoder and decoder. The encoder provides a sequence which has a fixed length and the decoder generates the correct sequence form the encoder sequence. For that in machine translation NMT provide a good result for short length sequence of text [8]. Since text summarization is new research in Bengali NLP different technique is invention day by day. This research work uses NMT for Bengali text summarization based on short Bengali text sequence. In our dataset text length is not big but good enough for summarization but the text summary is the short length within sentences. This research purpose we used NMT model with encoder and decoder. So short text are response a good result for text summarizer. Our algorithm also responses summary for a short sequence. But it is difficult to handle the long text sequences and summarize them. Thus there is another area of research to build a long or any length text summarization in Bengali text summarization.

2.5 Challenges

Structured Bengali data is not available. All data are present in an unstructured way. Therefore, data collection is a challenge for this research. Some newspaper dataset is available but some of the research work has almost done using this dataset. So, we need a new dataset to complete this research work. After the collection of dataset, the text data make a summary of that text is another challenging work. Working with the Bengali text is always challenging. The Library of NLTK is available for Bengali text processing. Therefore, in preprocessing step need to raw coding to prepare the text as an input of a model. Suppose, when removing punctuation from the text, need Unicode of each punctuation and remove it by raw coding. Another problem is stop word remove from the text. For other languages like English have a build-in library to remove stop words from text. But for the Bengali language collect stop word form online then put all in a text file and remove stop words from text using that file. A large vocabulary is another challenge is in this research. If the dataset has a large number of data, it would provide a large vocabulary and large vocabulary helps to generate an accurate summary.

CHAPTER 3

Research Methodology

3.1 Introduction

In this section, we will discuss about the whole methodology of the research work. Every research work has a unique solving technique. Applying all approaches are included in the methodology part. Here give a detailed discussion of applying using model with a short description of each individual parts of the methodology. In our research, we have used deep learning model for text summarization. The deep learning algorithms are used following the type of research contain. RNN is used for solving the text related problem in deep learning. Every deep learning model needs a good dataset to find an accurate automatic system. Before applying the algorithm dataset needs to be collected and preprocessed. In full part every section of methodology are discussed individually. Given all section are followed when the research work in completing. A better explanation of methodology increase the efficiency for work and give the nobility. Mathematical equation and graphical view of the model with there description is helping to understand the whole work. Therefore, further research and increasing the research filed good explanation of methodology is required.

Whole work looks like a framework. All steps of the methodology are briefly discussed in this chapter. Sub section of some core sections are helps to understand the gist of the model with it purpose of using. Working flow of whole research work is given below which give a short view of total research work.

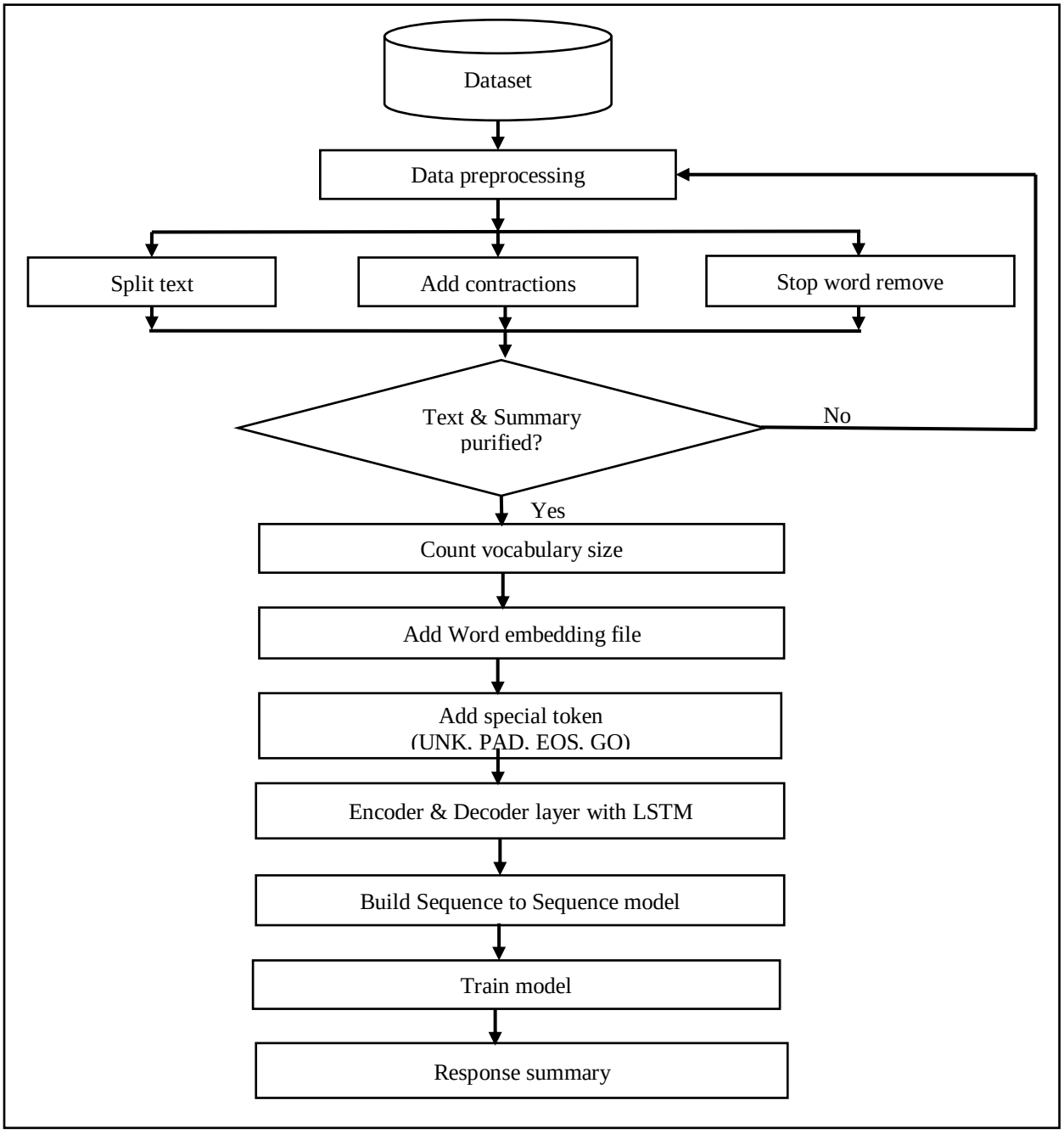


Figure 3.1.1 Workflow for text summarization

3.2 Research Subject and Instrumentation

Our thesis topic name is "Bengali Abstractive Text summarization using Deep Learning". This is a major research area in Bengali NLP. We have discussed the process of making an abstractive text summarization in Bengali with the conceptual and theoretical process first to now. A deep

learning model needs high configuration pc with GPU and others instrument. Now a list is given below of the required instrument for this model.

Hardware and Software:

- Intel Core i3 7th generation with 8GB RAM
- 1 TB HDD
- Google Colab with 12GB GPU and 350 GB RAM

Development Tools:

- Windows 10
- Python 3.7
- Tensorflow Backend Engine
- NLTK
- Pandas
- Numpy

3.3 Data collection and Data preprocessing

We used our own collected data for this research purpose. All data are collected from social media such as personal status, Blog posts, page posts and group posts. There is some complexity to collect data from social media for security issues. For that, we collect data using manual approaches. Total of 200 posts is collect form social media. After collection the post we have generated a summary of each post manually for the machine. Then dataset needs to preprocess and produce a clean text for the model. The necessary steps of preprocessing stage in the figure 3.3.1 are discussed section-wise in below.

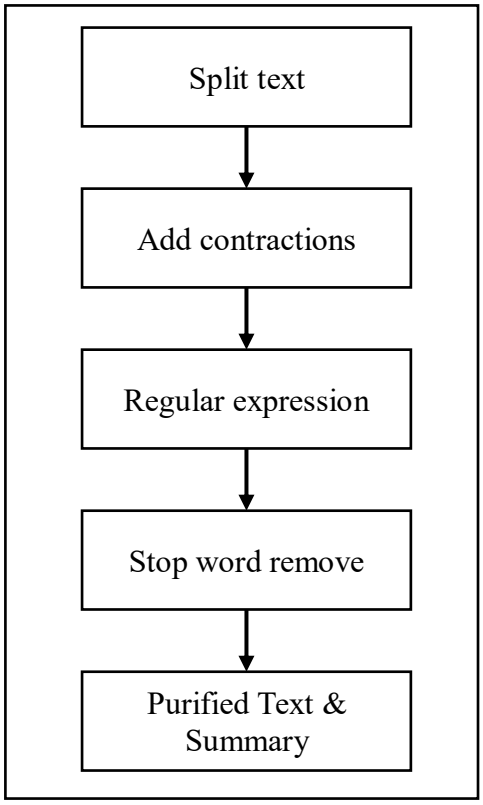


Figure 3.3.1 Dataset preprocessing

3.3.a Split text

After uploading the dataset whole text needed to be tokenized. Those are divided the long text to a single word. Which is helpful for clean text and remove unnecessary function form the dataset. Also, it helps create the vocabulary of the dataset which is important for NLP problems. This vocabulary is used to find relevant information form word embedding file.

3.3.b Add contractions

Each language has contractions of some words. Also, the Bengali language has few contractions for some word. It refers to the short form of a word or short writing technique of a word. The machine doesn't understand the short form of a text for that need to define the full meaning of the short form of the word. Some examples of Bengali contractions are "মো.": "মোহাম্মদ", "ইঞ্জি:": "ইঞ্জিনিয়ার" etc.

3.3.c Regular expression

The regular expression is used to remove the rare character or unwanted character to remove from the text. Remove space, whitespace, English character, punctuation form text, Bengali digit from text is the main use of the regular expression in our research.

3.3.d Stop word remove

Remove stop word text is a common process in NLP. Unnecessary word remove from the text is the main use of stop word. In NLTK build-in library produce to remove stop word from the English text. But there is no library available for Bengali stop word to remove. Therefore, at first, we collect all stop word in the Bengali language from online. There are total number of 393 stop words and then we inserted it into a file for further use. Some examples of stop words are, "কেউ", "খুবই", "অথচ", "ই" etc.

3.3.e Purified text & summary

After completing the previous steps, the sequence of the text and summary will look clean. Here all text and summary have not any punctuation or any extra space. All words look in a sequential order. Both clean text and summary are inserted into two different lists. Those lists are used input sequence of the summarization model. In table 1 an example of text preprocessing is given below.

Table 1: Example of text preprocessing

Original Text	Clean Text
আপনি ১২ বছর ইংরেজি পড়েও এক লাইন ঠিক মত লিখতে পারেন না, ভুলভাল করেও ১০০ শব্দ নিজের ভাষায় লিখতে পারেন না, ইংরেজিতে এক মিনিট কথা বলতে পারেন না। পারেন না দেখেই আমার গ্রুপে এসেছেন। আচ্ছা ঠিক আছে ১২ বছর যেখানে সময় দিয়েও পারেন না সেখানে আমার কথা মত ১২ মাস না, ১২ সপ্তাহও না মাত্র ১২ দিন সময় দিয়ে কমেন্ট করতে থাকেন। তাহলেই নিজের পরিবর্তন দেখবেন।	আপনি বছর ইংরেজি পড়েও এক লাইন ঠিক মত লিখতে পারেন না ভুলভাল করেও শব্দ নিজের ভাষায় লিখতে পারেন না ইংরেজিতে এক মিনিট কথা বলতে পারেন না পারেন না দেখেই আমার গ্রুপে এসেছেন আচ্ছা ঠিক আছে বছর যেখানে সময় দিয়েও পারেন না সেখানে আমার কথা মত মাস না সপ্তাহও না মাত্র দিন সময় দিয়ে কমেন্ট করতে থাকেন তাহলেই নিজের পরিবর্তন দেখবেন

3.4 Statistical Analysis

1. The total number of data 200. 200 data have 3 subsections such as Post type, Text and Summary. A short view of our dataset is given below in table 2.

Table 2: Sample of the dataset

Post Type	Text	Summary
personal	আন্দোলন ঠেকাতে সকল স্কুল-কলেজ বন্ধ করার ঘোষণা দিয়েছে কিন্তু এটা ভুলে গেছে যে আন্দোলন করার জন্য স্কুল-কলেজ খোলা থাকার দরকার পরে না।কেননা আন্দোলনটা হয় রাস্তায়।তাদের পন্থাটা এবার এমন হয়ে গেছে যে সাইক্লোন থেকে বাঁচার জন্য রেইনকোট পড়ে বসে আছে।কেউ আয়মান সাদিককে এদের কাছে পাঠিয়ে দে,১০ মিনিটে কিছু ট্রিক্স শিখিয়ে দিয়ে আসুক...	আন্দোলন করার জন্য স্কুল কলেজ বন্ধ করার দরকার হয় না।
pages	ঘূর্ণিঝড় ফণী আয়তনে বাংলাদেশের আয়তনের চেয়েও বড়, ২ লক্ষ বর্গকিলোমিটার। আঘাত হানলে সিডরের চেয়েও অনেক বড় আঘাত হবে। এটি একই যায়গায় বিগত ৫ দিন ধরে অবস্থান করে আরো শক্তিশালী হচ্ছে এবং হ্যারিকেনে রূপ নিচ্ছে। অবস্থা এত সিরিয়াস যে আবহাওয়া অধিদপ্তরের ছুটিতে থাকা স্টাফদের ছুটি বাতিল করা হয়েছে।	ঘূর্ণিঝড় ফণী আয়তনে বাংলাদেশের চেয়ে বড়।
group	আমি ইংরেজীতে দূর্বল...কিন্তু আমি স্বপ্ন দেখি ইংরেজীতে একদিন অনেক ভাল করব..কিন্তু কিভাবে বেসিক টু এডভান্স এ যেতে পারি?? কোথায় থেকে শুরু করতে পারি??	স্বপ্ন দেখি ইংরেজীতে একদিন অনেক ভাল করব।

2. The total size of the vocabulary is 5000k.
3. Use pre-trained word vector collected from online which have 497405-word embedding.
4. The total number of the unique word of the dataset is 4392k.
5. 94% of the word we used in our model.
6. Define the maximum length of the text is 83 words and the maximum summary length is 13 words.
7. Dataset is saved in excel file which extension is .xlsx.

3.5 Implementation Requirements

3.5.a. Problem discussion

In the dataset, the number of the text and their summary is equal. Generally, the text has a long length but the summary has a short length compared to each other. Now consider D contains the number of words of input text sequence of the dataset. Thus $x_1, x_2,\ldots$ is input sequence and which is coming from the vocabulary have to size V. That generates the output sequence such as $y_1,y_2,\ldots,y_d$, here S>D. That means the sequence of the summary is less than the input text document. Mention that all sequence is coming from a similar vocabulary.

3.5.b. Vocabulary and Word embedding

Counting the vocabulary is very important for word embedding. We count vocabulary based on the dataset. Then the vocabulary is used in a word embedding. For word, embedding needs a word vector. Here we used a pre-trained Bengali word vector file which was collected form online. Word to vector file contains a numeric value related word. It saves the words of all related word in a block. Then use the value when working. Those vector of the word is used as input of the model then provide related word which will be the output of the model. Therefore the sequence to sequence learning is easily complete its operation. We have used "bn_w2v_model" which is available online for everyone.

3.5.c. RNN Encoder & Decoder

After the invention of machine translation, a deep learning algorithm creates a great milestone in the Artificial Intelligence field. All text related problem are given accurate output in the deep learning model. RNN is the most usable algorithm in deep learning. It works more efficiently in any text related problem. Each RNN are made by LSTM cell. LSTM cell is like a short term memory. Encoder and decoder are used in LSTM cell. The input text is pass in the encoder where each input is word vector sequence. The decoder takes the input sequence and generates the output of the text from the relevant text sequence.

If we consider x is a target sequence of sentences than the maximum probability of the word vector sequence will be x. Here y is the source sequence of the sentences then probability will be,

$$\arg\left(max_y p(x|y)\right) \quad (1)$$

There are two types of RNN, one-directional and another is Bi-directional. One directional RNN has input and output each is connected with others in a sequential manner. Bi-directional RNN has two layers [9] with two directions. One is the forward direction and another is backward direction. All of those are used to solve the machine translation problem. In our work, we used two layered RNN. Since we used RNN for the Bengali language then fixed the length of Bengali is the input of the model which is carried by the encoder. Decoder gives the related sequence of the output depends on the input. Here the main calculation is working based on the probability calculation.

If X is the total input sequence where $X = (x_1, x_2, x_3 \dots \dots x_n)$ and if c is the context vector then the sequence will be,

$$h_t = f(x_t - h_{t-1}) \quad (2)$$

And c,

$$c = q(\{h_1, \dots . . h_{T_x}\}) \quad (3)$$

Here, At time t, h_t= hidden state and c = context vector. c is generated form h_t sequences. f and q both are non-linear functions.

If $y = \{y_1 \dots\dots, y_{T_y}\}$ is output sequence predicted by Decoder. Then the response or provided summary probability will be,

$$p(y) = \prod_{t=1}^{T} p(y|\{y_1, \dots, y_{t-1}\}, c) \dots\dots \text{(4)}$$

Here, $(y1, \dots .\, yTo)$ then the probability will be,

$$p(y|\{y_1, \dots, y_{t-1}\}, c) = g(y_{t-1}, s_t, c) \dots\dots \text{(5)}$$

Here, g = non-linear function, = output of probability, s= hidden sate.

$$c_i = \sum_{j=0}^{T} a_{ij}\, h_j \dots\dots \text{(6)}$$

Suppose, forward layer the input sequence $(x_T\ to\ x_{T_x})$, also $(\overrightarrow{h_1}\ to\ \overrightarrow{h_{T_x}})$ is the hidden state. Then the backward layer input sequence is $(x_{T_x}\ to\ x_T)$ and the hidden state $(\overleftarrow{h_{T_x}}\ to\ \overleftarrow{h_1})$ thus,

$$h_j = [\, \overrightarrow{h_{jT}}; \overleftarrow{h_{jT}}\,]^T \dots\dots \text{(7)}$$

Here, h_j= predicted summary and succeeding word.

$$e_{ij} = a(s_{i-1}, h_j) \dots\dots \text{(8)}$$

Now a graphical view of the model is given below,

All of those eqations are the explanation of using model and helps to undrstand to the voew point of whole reseach using this approaches.Graphical view of the main model is given below for ths research methodology.

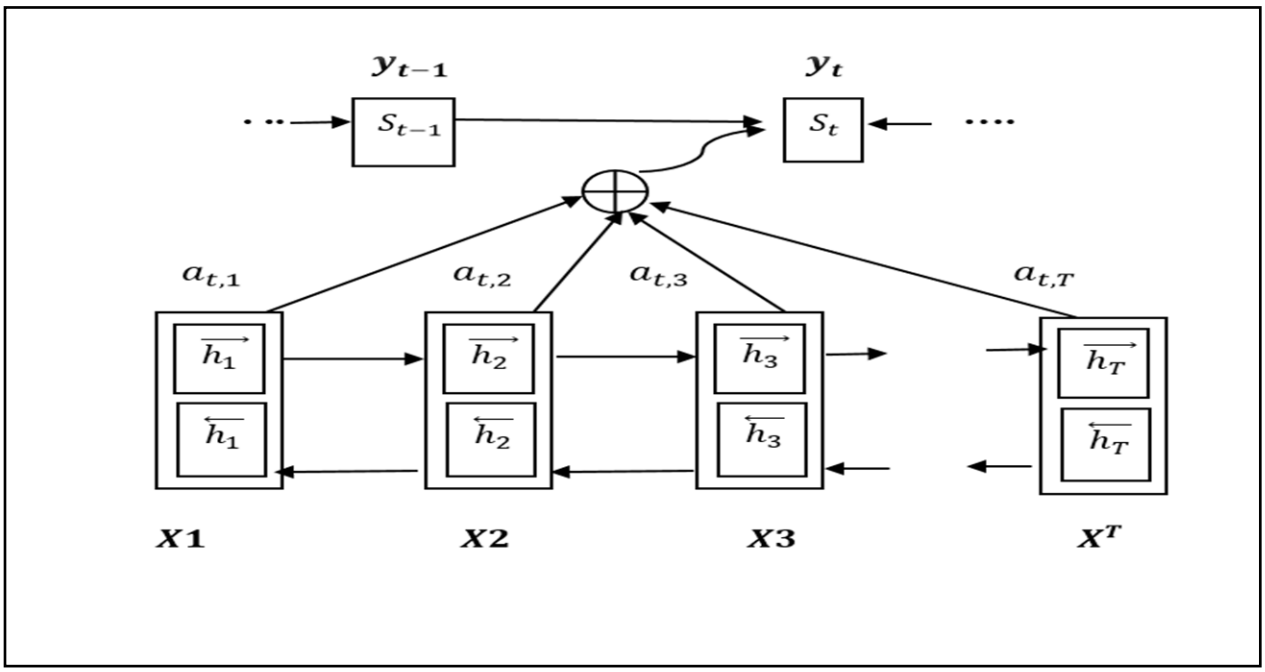


Figure 3.5.1 Graphical View of the model

3.5.d. Sequence to Sequence Learning

Seq2Seq model is created by LSTM cell. Firstly, the input of the word is formed from the vector file. In the vector file, each related word has an embedded value. Those embedded values are worked like the input of the encoder. The encoder saves the sequence value in short memory which is LSTM. Here each sequence used a token to identify the end and start point of the sequence. In the program, we defined some special sequence such as <PAD>, <EOS>, <GO>, <UNK> etc. All of those special tokens are used for working in handling the sequence in the encoder and decoder. <EOS> is used to identify the end of the input sequence. In the encoder when the sequence of the input ends the <EOS> token automatic discard the sequence. Then the sequence will go to the decoder to decode the sequence by providing related output. End of the decoder that means when the output sequence ends the <EOS> token stop the decoder. Figure 3.3.3 shows the working process of the encoder and decoder. After the end of the encoding, the sequence needs an instruction to enter the decoder. Here we use <GO> token to give the instruction of encoding sequence to enter the decoder.

In the text sequence, some the text or word are not replaced. All of that sequence need to identify. Therefore, we used a special token <UNK> which means an unknown token. When an unknown token is found in the sequence it will be added <UNK> token in the text. In the train, the time sequence is divided into the batch. In a batch size similar length of the sequence needed to be together. Thus we used a token which is known as <PAD> token.

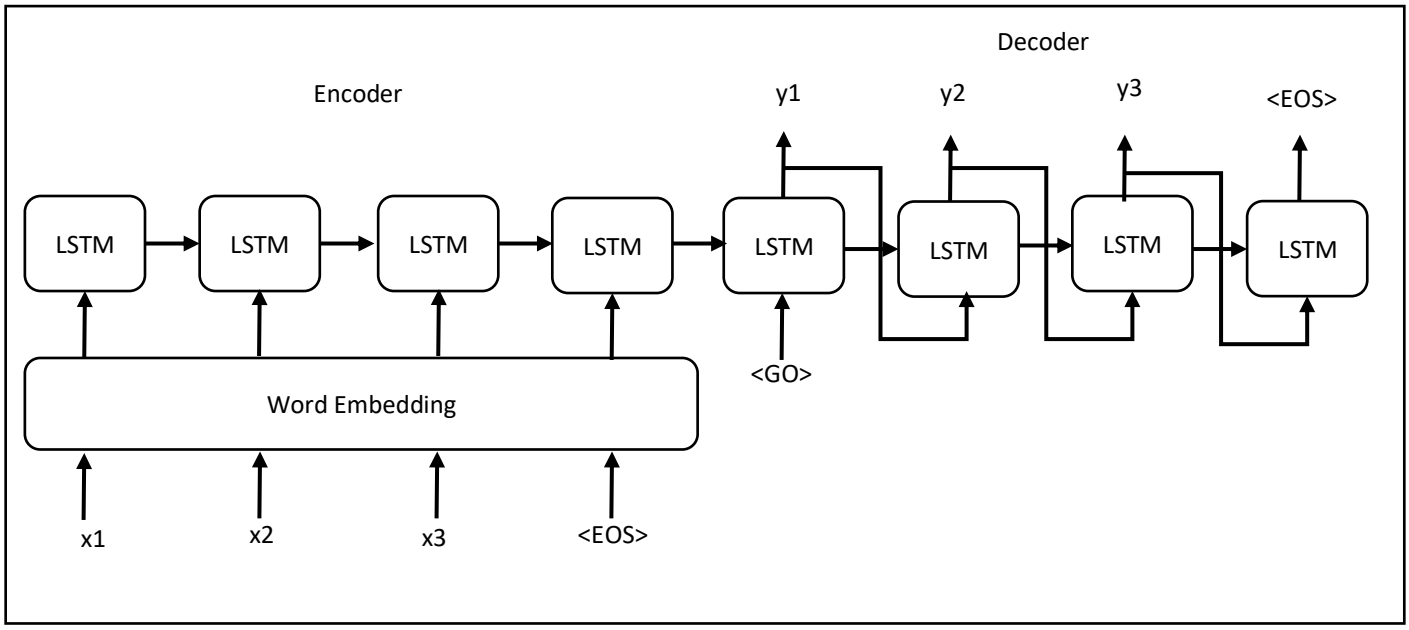


Figure 3.3.3: View of Seq2Seq model

The above figure describes the Seq2Seq model process of the sequence. Also given a view of how a long sequence can provide a probable output sequence. In the encoder, Bengali text is the input of the model and summary of the corresponding sequence is output provided by the decoder.

CHAPTER 4

Experimental Results and Discussion

4.1 Introduction

Abstractive text summarization is a complex problem in NLP. The machine can summarize the text automatically that is not mandatory for response summary are present or not present in the text. Therefore, find an accurate summary is difficult. Probability calculation is important for this text summarizer. Because the machine gives output on the basis of the maximum probability. In the model weight of each word are learn in the train time then calculate the probability response the summary based on the related word weight. After the preprocessing, the text data need to train for the model to learn the machine. For training, each deep learning model has the backend engine. This experiment we used TensorFlow 1.15.0 for a backend engine. For the train, some basic parameter value needs to define. Such as epochs, batch size, learning rate, number of layer etc. All of those parameter training is dependent. Reduce the loss in train time is very important. In this experiment we have used "Adam" optimizer to reduce the loss and optimize the model. A well-trained model can give a batter output in the test time. High configuration pc needs to train data in the deep learning model. GPU is very helpful for that.

This experiment we have not to build in GPU to train our model. For that, we train our model firstly in direct pc. That takes huge time to train the model and the given output is not satisfactory for summarizer text. Finally, we train the model using google colab. Which provides free GPU service for the user. That is very first and reduce train time. Now the value of the parameter is given below which is using this experiment.

- Define Epochs = 70
- Define Batch size = 2
- Set the RNN cell size = 256
- Set the number of neuron layer = 3
- Define the model Learning rate= 0.001
- Keep the probability rate is = 0.70 to 0.75

4.2 Experimental Results

The machine gives output nearly the actual output. Everybody knows that no machine gives an 100% accurate output. Similarly, our trained model gives a good result but not for all values. Sometimes it responses wrong text corresponding to the text. But the maximum number of response words is similar to the meaning of the text.

We train our model in 70 epochs then reduce the loss which is 0.008. For checking the output, we save the model in a file whose name is “model.ckpt” file. Then we create TensorFlow session for reloading the graph which was saved in previous steps. Then define the text and summary data frame randomly to check their summary. After that, we convert the in value to vocabulary for the sequence which was the input of the model.

Previously we have created a logistic function to give the response answer logically. The logistic function is the response to the summary based on the probability. The probability value is calculated by the weight value and embedding value of the text. Two sample output of our result is given below in table 3 and table 4. Each table original text contains the raw data which was collected form online. Original summary are provided by a human for corresponding text. After preprocessing of raw text, text are converted into pure text which is Input words. Final variable response word given by machine after the training and learning.

Table 3: Sample example one of the response summary

Original Text:	আপনি ১২ বছর ইংরেজি পড়েও এক লাইন ঠিক মত লিখতে পারেন না, ভুলভাল করেও ১০০ শব্দ নিজের ভাষায় লিখতে পারেন না, ইংরেজিতে এক মিনিট কথা বলতে পারেন না। পারেন না দেখেই আমার গ্রুপে এসেছেন। আচ্ছা ঠিক আছে ১২ বছর যেখানে সময় দিয়েও পারেন না সেখানে আমার কথা মত ১২ মাস না, ১২ সপ্তাহও না মাত্র ১২ দিন সময় দিয়ে কমেন্ট করতে থাকেন। তাহলেই নিজের পরিবর্তন দেখবেন।
Original Summary:	পারেন না দেখেই আমার গ্রুপে এসেছেন।
Input Words:	আপনি বছর ইংরেজি পড়েও এক লাইন ঠিক মত লিখতে পারেন না ভুলভাল করেও শব্দ নিজের ভাষায় লিখতে পারেন না ইংরেজিতে এক মিনিট কথা বলতে পারেন না পারেন না দেখেই আমার গ্রুপে এসেছেন আচ্ছা ঠিক আছে বছর যেখানে সময় দিয়েও পারেন না সেখানে আমার কথা মত মাস না সপ্তাহও না মাত্র দিন সময় দিয়ে কমেন্ট করতে থাকেন তাহলেই নিজের পরিবর্তন দেখবেন
Response Summary:	পারেন না দেখেই আমার গ্রুপে

Table 4: Sample example two of the response summary

Original Text:	এই ওরে মার, সবাই চারদিক দিক থেকে ঘিরে মার। ভাই ওর হাতে পাথর নাই, পাথর নাই ? তাইলে ওরে বাঁচয়ে রেখে লাভ কি ? কাটাকাটি করে মেরে ফেল।পাথরের হতাশায় ৫ বছর চলে গেল। একজনের একটা ৫ বছরের মেয়েও আছে। অন্যদিকে গাড়ির ভিতরে ইদুর কি করতে করতে সুইচে চাপ পরে গেছে। অতীতে গিয়ে বর্তমান পরিবর্তনের একটা হাউ কাউ আছে... অতীতে গিয়ে তারা পাথর খুঁজে এনে আবার মারামারির প্রস্তুতি নিচ্ছে...
Original Summary:	অতীতে গিয়ে তারা পাথর খুঁজে এনে আবার মারামারির প্রস্তুতি নিচ্ছে।
Input Words:	এই ওরে মার সবাই চারদিক দিক থেকে ঘিরে মার ভাই ওর হাতে পাথর নাই পাথর নাই তাইলে ওরে <UNK> রেখে লাভ কি কাটাকাটি করে মেরে ফেল পাথরের হতাশায় বছর চলে গেল একজনের একটা বছরের মেয়েও আছে অন্যদিকে গাড়ির ভিতরে ইদুর কি করতে করতে সুইচে চাপ পরে গেছে অতীতে গিয়ে বর্তমান পরিবর্তনের একটা হাউ কাউ আছে অতীতে গিয়ে তারা পাথর খুঁজে এনে আবার মারামারির প্রস্তুতি নিচ্ছে
Response Summary:	অতীতে গিয়ে তারা পাথর খুঁজে এনে আবার

4.3 Descriptive Analysis

Before the making of Bengali text summarization model, we have created a model for English text summarization. Both models give a good result for different text. Summarization model is created to reduce the loss of function in the model. The loss function evaluates the model. It decreases the error for learning model. Loss function reduce is very important for sequential data. In train time we have calculated the loss function. After completing the iteration in train time final loss function is calculated. At first, when the model started the loss is very high. But in the final stage, the loss is reducing the weight value. We found the weight value from the experiment which is 0.008. Before trained model, we have divided the data for train and test. We have set the values 180 for train data and 20 is for the test data with the summary.

4.4 Summary

This section discussed the experiment of our model. And the response of the machine to create a summary. All are discussed in upper briefly in details with the demon of machine response summary.

CHAPTER 5

Conclusion and Future Work

5.1 Summary of the Study

Our whole project is related to the Bengali NLP. In this project, we have built a deep learning model for Bengali abstractive text summarization. That is very helpful for making an automatic Bengali text summarization. We have completed this project within the 3 months. The whole project is divided into some parts. The whole summary of the project is given below with step by step.

Step 1: Data collection form social media

Step 2: Summarize the collected data

Step 3: Collect word2vec

Step 4: Data preprocessing

Step 5: Vocabulary count

Step 6: Load pre-trained word2vec

Step 7: Add special token

Step 8: Define Encoder and Decoder with LSTM

Step 9: Build sequence to sequence model

Step 10: Train model

Step 11: Check the result analysis the response of the machine

This model will help our Bengali NLP research community to build a full dependent automatic abstractive text summarization and further research of Bengali text summarization. Now we will discuss the future work and conclusion of this research work.

5.2 Conclusion

The main concern of this research work is developing and increasing the Bengali NLP research area. We have used the Bengali text as input of our model and generated a summary of those sequence will be Bengali text which is the output of the model. At first, we build the model for English text then we make this model for Bengali text. Normally encoder and decoder are working similarly for both texts similarly. Our dataset is not large for Bengali text. But the machine provides excellent responses for this dataset. This model is built for the Bengali short text summary generation. We have defined the sequence length and summary length. The machine can generate the summary following this fixed length. This is the main limitation of our model. If the sequence length is over the model does not work properly. This is the another research scope for Bengali text summarization. Bengali text preprocessing is a little bit difficult on the oppose to another language. Therefore, preprocessing library needs to build for Bengali text. Word to vector is another important part of these kinds of problem. Strong word to vector needs to produce for solving text related problem. After all, no machine gives fully accurate result. Every machine has some limitation in their working filed. Similarly, our summarizer model also has some limitations. But the main thing is that the model can generate an abstractive summary for the Bengali language. This is an achievement for our Bengali NLP filed which helpful for future research work.

5.3 Recommendations

In the next stage of our work we will increase the dataset and their summary for improving the model performance. We will try to build another model for text summarization which will help us to find the best performer for the Bengali text summarization. We are working only the short sequence but for long text sequence, a summarizer is needed in the Bengali language. Some recommendations for text summarization is given below,

- Understand the abstract of long text
- Reduce the reading time
- Reduce the document size with saving the core information
- Automatic system to extract information

5.4 Implication for Further Study

Some limitation is presented in this is model such as work for limited sequence, the dataset is not enough. But the model is built for future development. Since any research work is a continuous process. Therefore, this model will be developed day by day for the Bengali language. To find a proper solution any works need more research. Then all research find a proper solution for a specific problem. So, research work needs future implement or development. The future implement is dependent on the limitations of the previous work. Solving the limitations of the previous work helps to make an efficient system. In this work, the future work will be increasing the dataset of the Bengali text. Updating the model and prepare the model of any kinds of text length. That means the model won't dependent on the text length. The model is complex and working in TensorFlow 1.15 version. But need to convert the code in updated versions. After completing research the model needs to deploy. Thus, making an application like web and mobile application is important based on the future of artificial intelligence. Therefore, we have developed an application for automatic Bengali abstractive text summarization.

REFERENCES

[1] Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate." arXiv preprint arXiv:1409.0473 (2014).
[2] Luong, Minh-Thang, Hieu Pham, and Christopher D. Manning. "Effective approaches to attention-based neural machine translation." arXiv preprint arXiv:1508.04025 (2015).
[3] Wu Y, Schuster M, Chen Z, Le QV, Norouzi M, Macherey W, Krikun M, Cao Y, Gao Q, Macherey K, Klingner J. Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144. 2016 Sep 26.
[4] Nallapati, Ramesh, et al. "Abstractive text summarization using sequence-to-sequence rnns and beyond." arXiv preprint arXiv:1602.06023 (2016).
[5] Li, Piji, et al. "Deep recurrent generative decoder for abstractive text summarization." arXiv preprint arXiv:1708.00625 (2017).
[6] Wang, Li, Junlin Yao, Yunzhe Tao, Li Zhong, Wei Liu, and Qiang Du. "A reinforced topic-aware convolutional sequence-to-sequence model for abstractive text summarization." arXiv preprint arXiv:1805.03616 (2018).
[7] Gehring, Jonas, et al. "Convolutional sequence to sequence learning." Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017.
[8] K.Cho, B .van Merrienboer, D.Bahdanau, Y.Bengio " On the Properties of Neural Machine translation: EncoderDecoder Approaches". Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8),7oct 2014.
[9] Cho, K. et al. (2014) Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. Proceeding softhe2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)
[10] Sutskever et al "Sequence to Sequence Learning with Neural Networks". Conference on Neural Information Processing Systems (NIPS,2014).
[11] Peter J. Liu et al. "Generating Wikipedia by Summarizing Long Sequences". International Conference on Learning Representation (ICLR), 2018.
[12] Lifeng Shang, Zhengdong Lu, Hang Li "Neural Responding Machine for Short-Text Conversation". Association for Computational Linguistics (ACL 2015)
[13] H. T. Le and T. M. Le, "An approach to abstractive text summarization," 2013 International Conference on Soft Computing and Pattern Recognition (SoCPaR), Hanoi, 2013, pp. 371-376.
[14] I. F. Moawad and M. Aref, "Semantic graph reduction approach for abstractive Text Summarization," 2012 Seventh International Conference on Computer Engineering & Systems (ICCES), Cairo, 2012, pp. 132-138.
[15] D. Sahoo, A. Bhoi, and R. C. Balabantaray, "Hybrid Approach To Abstractive Summarization," Procedia Computer Science, vol. 132, pp. 1228–1237, 2018.
[16] Ganesan, K., Zhai, C., Han, J. 2010. Opinosis: A Graph Based Approach to Abstractive Summarization of Highly Redundant Opinions. In Proc. of Coling 2010, pages 340–348.
[17] Lloret, E., Palomar, M. 2011. Analyzing the Use of Word Graphs for Abstractive Text Summarization. In Proc. of IMMM 2011.

Bengali Abstractive Text Summarization Using Deep Learning

ORIGINALITY REPORT

9%	5%	2%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	4%
2	Submitted to University College London Student Paper	<1%
3	"Natural Language Processing and Chinese Computing", Springer Science and Business Media LLC, 2018 Publication	<1%
4	"Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data", Springer Nature, 2016 Publication	<1%
5	arxiv.org Internet Source	<1%
6	www.hangli-hl.com Internet Source	<1%
7	dspace.jaist.ac.jp Internet Source	<1%

Submitted to Rivers State University of Science

8	& Technology Student Paper	<1%
9	"Natural Language Processing and Chinese Computing", Springer Nature America, Inc, 2018 Publication	<1%
10	link.springer.com Internet Source	<1%
11	projectcare.com.ng Internet Source	<1%
12	Luo, Jing, Bo Meng, Changqin Quan, and Xinhui Tu. "Exploiting salient semantic analysis for information retrieval", Enterprise Information Systems, 2015. Publication	<1%
13	trap.ncirl.ie Internet Source	<1%
14	Submitted to Savitribai Phule Pune University Student Paper	<1%
15	Trung Tran, Dang Tuan Nguyen. "Text Generation from Abstract Semantic Representation for Summarizing Vietnamese Paragraphs Having Co-references", 2018 5th NAFOSTED Conference on Information and Computer Science (NICS), 2018 Publication	<1%

16	hdl.handle.net Internet Source	<1%
17	Submitted to University of Southampton Student Paper	<1%
18	Submitted to Radboud Universiteit Nijmegen Student Paper	<1%
19	Submitted to Aristotle University of Thessaloniki Student Paper	<1%
20	docplayer.net Internet Source	<1%
21	aclweb.org Internet Source	<1%
22	Submitted to Coventry University Student Paper	<1%
23	export.arxiv.org Internet Source	<1%
24	Jinpeng Li, Chuang Zhang, Xiaojun Chen, Yanan Cao, Pengcheng Liao, Peng Zhang. "Abstractive Text Summarization with Multi-Head Attention", 2019 International Joint Conference on Neural Networks (IJCNN), 2019 Publication	<1%
25	www.utupub.fi Internet Source	<1%

26	d-nb.info Internet Source	<1%
27	e-spacio.uned.es Internet Source	<1%
28	dspace.lboro.ac.uk Internet Source	<1%
29	doras.dcu.ie Internet Source	<1%
30	"Natural Language Processing and Chinese Computing", Springer Science and Business Media LLC, 2019 Publication	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		