ABSTRACTIVE TEXT SUMMARIZATION USING BIDIRECTIONAL LSTM WITH ATTENTION MECHANISM AND WORD2VEC

Manav Jain[1], Riddhi Shah[2], Devansh Mehta[3], Vinaya Sawant[4]

[1,2,3]Department Of Information Technology, Dwarkadas J. Sanghvi College of Engineering, Mumbai, Maharashtra, India.

[4]Head Of Department, Department of Information Technology, Dwarkadas J. Sanghvi College of Engineering, Mumbai, Maharashtra, India.

Abstract. Automatic text summarization is an approach to extract pivotal information from a large source of text. Attention mechanism enables to highlight important and relevant features of the input data, in Natural Language Processing(NLP). It is basically a sequence of textual elements. Long short-term memory (LSTM) with attention mechanism, Bidirectional LSTM with attention mechanism, and Bidirectional LSTM with attention mechanism along with one Word2Vec Embedding layer are the three models on which we trained our dataset. The main aim of this paper is to evaluate and compare the performance of all three models using BLEU and ROUGE score. This paper also suggests the reason as to why one of the models performs better than others in BLEU score and performs relatively poor in ROUGE score. Further, the performance of the model bidirectional LSTM with Attention Mechanism and Word2Vec Embedding Layer is improved by using Beam Search Decoder instead of Argument max that is Greedy search algorithm.

Keywords- NLP, Seq2Seq, LSTM, Bidirectional LSTM, Attention mechanism, BLEU, ROUGE, Word2Vec.

1. Introduction

Text summarization refers to the procedure of compressing long bits of text into smaller text but it makes sure that all the pertinent information does not get lost while compressing it [10]. The main objective is to extract some meaningful and important features out of the input data. Today, Automatic Text summarization has become a common problem in Natural Language Processing. With the tremendous amount of data circulating in the digital world, there is a need to develop a model which automatically reduces long text and generates summaries which can help understand intended messages [11].

Summarization has two main approaches, one is Abstractive summarization where it trains itself and paraphrases the whole document into its short own version of summary and other one is Extractive Summarization which extracts meaningful and important sentences from the article without any modifications[8]. Text Summarization has a plethora of real-world applications such as Internal document workflow, Legal contract analysis, Question answering and bots, E-learning and class assignments, Help desk and customer support, Automated content creation and many more.

Abstractive summarization method aims at generating outlines by interpreting the text document using advanced natural language techniques in order to produce a new **shorter** summary which is not the copy of the original document [9]. It conveys only crucial information from the original text. There are three main models which can be used such as Neural Machine Translation (NMT) which is use to predict the likelihood of a sequence of words, Seq2Seq model which takes a sequence of items and outputs another sequence of words and Word2Vec Embedding that is used to learn word associations from a large corpus of text [1][2].

Our main goal is to build a text summarizer where the input is Reviews from the dataset and the output is a summary. Both are long sequences of words . Thus, we modeled this problem as a Many-to-Many Seq2Seq problem. The Seq2Seq model consists of two major components:

1) Encoder: The input data that is the embedded matrix is fed into the encoder where the embedding layer of the encoder contains weight. The encoder states is a Bidirectional LSTM that is the concatenation of forward and backward states along with encoder output.

2) Decoder: The encoder state is used to initialize the decoder where the decoder contains its own weight. The unidirectional LSTM decoder now contains the latent dimensions. The output of the decoder is then concatenated with the attention layer for further modeling.

The encoder and decoder architecture is used in the training as well as inference phase. However, there are limitations in the encoder-decoder architecture. To overcome these limitations, Attention Mechanism is used.

2. Literature Survey

The Impact of Local Attention in LSTM for Abstractive Text Summarization [3], this paper proposes a system which contains a subset of input words rather than whole input words. Amazon Fine Food Reviews dataset is used to build the model where the data is processed only between 25 to 300 characters for each text [13]. The unwanted characters and symbols are removed and also the data is converted into lowercase in the cleaning step. Further, tokenization is applied where data is converted into tokens for each text and summary. Next step is to apply the GloVe vector to word embeddings and the words not registered in the GloVe vectors are expressed as <UNK>. The last step is batch distribution where the highest number of tokens in the text is chosen as reference and a <PAD> tag is added to each and every text to make sure each array batch is of the same size. Once the preprocessing is done, an LSTM model is applied to combine the old states with the new ones. The LSTM model contains an encoder in which the forward encoder reads the vectors from the front and the backward encoder reads the sequence vector from behind. The decoder converts the vectors into corresponding English words. Three different models are used by changing the parameters where the first model uses Global attention and the rest of the models use Local attention. After evaluating these models it turned out that the local attention model performed better by producing two word pairs or more. Thus, the author states that window length has an impact in producing result text.

Another approach for text summarization was suggested in the paper Abstractive method of text summarization with sequence to sequence RNNs [4]. This paper focuses on summarizing english to english text. The author has used Amazon Fine Food Reviews [13] dataset which contains 568454 reviews but out of those they have only selected 20000 reviews. The author has built the model using three phases: data preprocessing, encoder-decoder layer for LSTM, sequence to sequence model. In the data preprocessing stage stop words are used to keep only essential information. After that GloVe vector is used for word to vec files. In the second stage, the 2 layer RNN's encoder decoder model is used where the encoder contains sentences of fixed length as input and decoder contains the output as sequence. The hidden units are used to increase memory capacity and training. Further Neural machine translation is incorporated to maximize the conditional probability. In the final stage, sequence to sequence model is used to train the model where the model works exceptionally well for the smaller texts but it's less efficient for the larger texts.

Further, Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond[5] suggest a methodology which uses bidirectional GRU-RNN for encoder and unidirectional GRU-RNN for decoder. The decoder consists of a switch which acts as a generator that produces words from target vocabulary when turned on, and acts as a pointer which points to the word position when turned off. Sigmoid activation function is used for the switch. The author has performed the experiments in two corpuses. First is the Gigawords corpus where 2000 examples were used for testing and validating. The training model consisted of a hidden state with a fixed dimension of 400. The learning rate was kept as 0.001 with a batch size of 50 and decoder vocabulary size was restricted to 2000 for better performance. The model outperformed ABS+ model which was tuned on DUC corpus. The second corpus used was CNN/Daily Mail corpus, which was produced by modifying the existing corpus. The modification of the script led to restoring bullet summary for each story where each bullet represents a sentence. This dataset is relatively smaller and more complex than Gigawords corpus still it produced a Rouge score similar to that of the Gigawords dataset.

3. Developed System

The proposed model is to produce abstractive text summarization of the given text using Bidirectional LSTM and Global Attention Mechanism as illustrated in Figure 1. The detailed descriptions are given in subsections below.

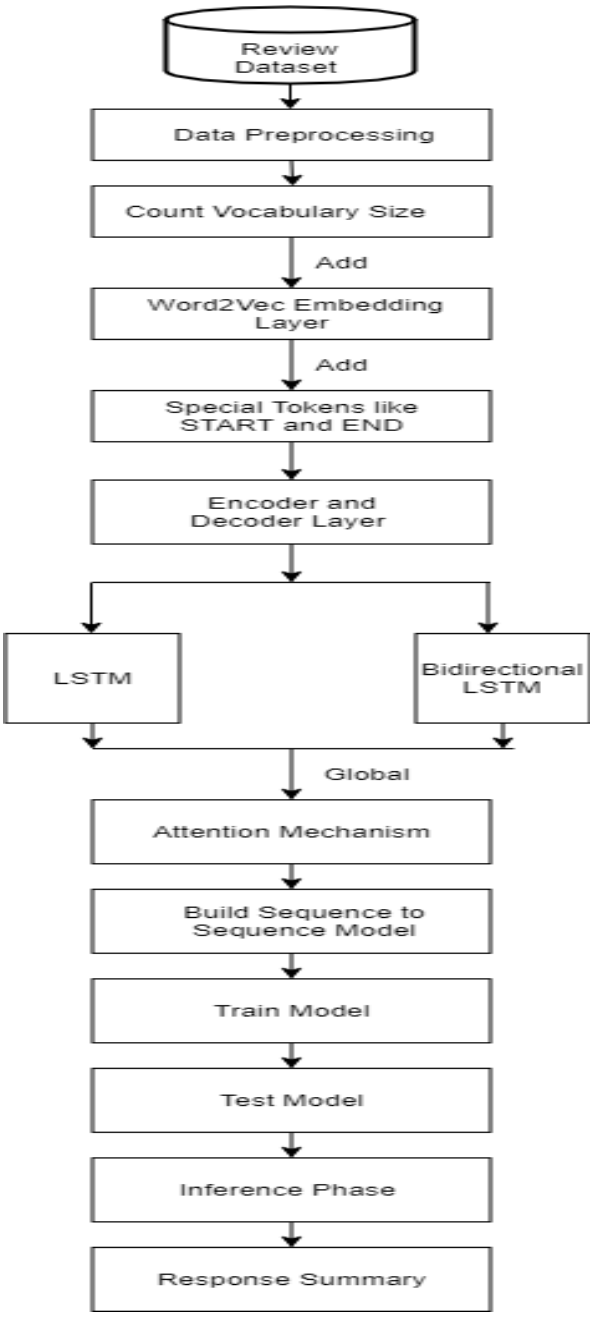

figure 1. system architecture

3.1 Dataset

The dataset used was "Amazon Fine Food Reviews [13]" to build the model. This dataset contains reviews of fine foods from amazon [13]. It consists of more than ~5,00,000 reviews from the last ten years [13]. Reviews include user information, product data, ratings, summary and a plain text review [13]. It also includes reviews from all other Amazon categories [13]. We have taken 1,00,000 reviews to reduce the training time of our model. We also have considered the columns of Summary and Text explicitly and removed other columns for building our model which is shown in figure 2.

	Summary	Text
0	Good Quality Dog Food	I have bought several of the Vitality canned dog food products and have found them all to be of good quality. The product looks more like a stew than a processed meat and it smells better. My Labr...
1	Not as Advertised	Product arrived labeled as Jumbo Salted Peanuts...the peanuts were actually small sized unsalted. Not sure if this was an error or if the vendor intended to represent the product as "Jumbo".
2	"Delight" says it all	This is a confection that has been around a few centuries. It is a light, pillowy citrus gelatin with nuts - in this case Filberts. And it is cut into tiny squares and then liberally coated with ...

figure 2. screenshot of the first three rows of the dataset.

The next dataset is GloVe(Global Vectors for Word Representation), which is used for word embedding in the embedded layer of encoder as well as decoder .It is a Word2Vec dataset to convert words into each vector shape[4]. It is a word vector with 100d, where each word is converted into a one-hundred-dimensional vector [4].

3.2 Data Pre-Processing

In this phase ,we performed some tasks on data. Firstly, we remove the null values and drop the duplicates then we converted each and every text to lowercase .We removed any HTML tags in the text and split the text .There are many contractions in English Language such as didn't ,doesn't ,haven't, I'd etc .So, we added contraction mapping in pre-processing stage . We now worked on cleaning the text, where we removed the unnecessary things like - 's , parenthesis , punctuations and special characters .Next step , we remove stop words which contain little to no unique information. After stop words , we lemmatize the words, grouping together different variants of a word so that they can be analyzed as one .After completing all the steps ,we get cleaned text . Now these pre-processing steps are applied to both the reviews and summaries in the dataset and then we obtained clean reviews and summaries which we used for building our model .We also included the '<sos> tok' and '<eos> tok' special tokens signs at the beginning and end of the cleaned summary. These tokens are added to the target sequence, so that the decoder knows where the sentence starts and ends. The preprocessed data is shown in figure 3.

	summary	text
0	sostok good quality dog food eostok	bought several Vitality canned dog food products found good quality. The product looks like stew processed meat smells better. Labrador finicky appreciates product better most.
1	sostok not as advertised eostok	Product arrived labeled Jumbo Salted Peanuts...the peanuts actually small sized unsalted. Not sure error vendor intended represent product "Jumbo".
2	sostok delight says it all eostok	This confection around centuries. light, pillowy citrus gelatin nuts case Filberts. And cut tiny squares liberally coated powdered sugar. And tiny mouthful heaven. Not chewy, flavorful. highly rec...

figure 3. screenshot of the first three rows of pre-processed data

3.3 Understanding distribution of sequences

Here, we analyze cleaned reviews and summaries to get a comprehensive idea about the length's distribution of the text which is depicted in Figure 4. By taking help of this, we find the sequence with the maximum length .We find out that 97% of the summaries have length below 10 and and 95% of reviews have length below 100. Thus, we fix the maximum length of the reviews to 100. Similarly, we set the maximum length of the summary to 10.

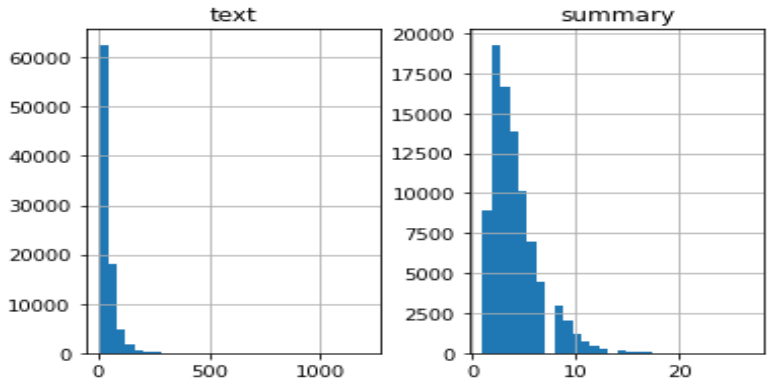

figure 4. graphical analysis for the cleaned review and summaries

3.4 Tokenization and Word Embedding

Tokenizer builds the vocabulary and converts a word sequence to an integer sequence[12].We used a tokenizer for review and summary[12]. We considered the rare words proportion and its total coverage in the whole text[12]. The threshold for review was set to be 4 which means a word whose count is lower than 4 is considered to be a rare word .Similarly, the threshold for summary was set to be 6 which means a word whose count is lower than 6 is considered to be a rare word. In this ,it uses the size of the vocabulary which means every unique word in the text.It then takes into consideration the no of rare words whose count falls below threshold and counts them. Padding of zero is done upto maximum length for uniformity in tokenization .It gives the top most common words.

We used a pre-trained word2vec file to improve our models accuracy. There are several available pre-trained word2vec files such as GloVe , ConceptNet Numberbatch , wiki-news-300d-1M.vec,crawl-300d-2M.vec etc[4]. In this project, we have used GloVe for word embedding in the embedding layer to build the model.

3.5 Model

- Bidirectional LSTM: Bidirectional recurrent neural networks (RNN) are simply assembling two independent RNNs. This structure permits the organizations to have both in reverse and forward data about the succession at each time step. Using two LSTMs instead of one, the output layer that will be produced can get information from backward and forward states simultaneously. For forward go, forward states and in reverse states are passed first, at that point output neurons are passed. In reverse pass, output neurons are passed first, at that point forward states and reverse states are passed next. After forward and in reverse passes are done, the weights are updated.

- Attention Mechanism: Attention mechanism is used in a way that it memorizes long source sentences in the context of Neural Machine Translation using Seq2Seq Models. An Encoder LSTM operates on input sentences and encodes it into a fixed-length vector. A Decoder LSTM produces an output word by translation from the encoded vector[5].Attention Mechanism permits the decoder to take care of various pieces of the source sentence at each progression of the output generation[7]. Instead of encoding the complete input sentence into a fixed context vector, the model automatically produces a context vector for each and every output time step. The model learns two things, one is what to attend based on input sentence and other one is what it has created until that point. The architecture of global attention mechanism is illustrated in figure 5.

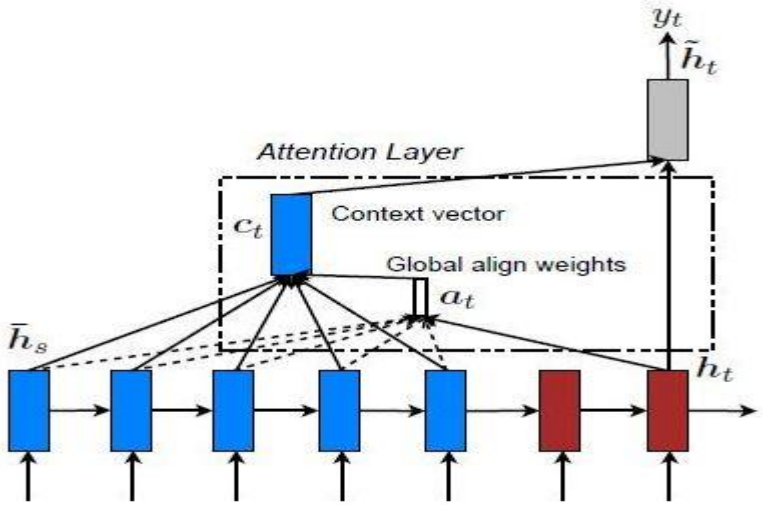

figure 5. global attention architecture

- LSTM: LSTM is an artificial Recurrent Neural Network which has the ability to learn long term dependencies in problems associated with sequence predictions [7]. It can be used to process both the single points(for example images) and the entire sequence of data(such as speech, video)[8]. It was designed in such a way that it helps models which have failed to operate when only RNN was used. It has a wide range of applications where LSTM comes into picture such as Text generation, Image processing, Language translation and Speech or handwriting recognition, etc.

4. Experiment and Output

We have performed the text summarization using Attention Mechanism on LSTM and Bidirectional LSTM. Sparse categorical cross-entropy is used as the loss function since it converts the integer sequence to a one-hot vector on the fly. Any memories issue is solved through this. The idea behind early stopping is also used to stop the training of neural networks by monitoring the validation loss(val_loss). The model training will stop once there is an increase in validation loss. The model was trained on a batch size of 128 and 90% dataset was used for training and 10% dataset was used for validation .In text summarization using Attention mechanism on Bidirectional LSTM ,we build a 3 stacked bidirectional LSTM for the encoder. The embedding layer of the encoder contains weight that is the input embedded matrix which is the embedding vector created by word2vec.Similarly , the embedding layer of the decoder also contains weight .The outputs from the bidirectional lstm consist of encoder output, forward state_h, backward state_h , forward state_c , backward state_c .The forward and the backward state for c and h are concatenated to form the encoder states .This encoder state is then used to initialize the decoder lstm .Now decoder lstm has a LSTM layer with latent dimensions double the bidirectional LSTM. An attention layer which implements Bahdanau Attention is used to capture the context of important parts of the source sequence that result in the target sequence . The attention layer's output is concatenated with the decoder's output . The output is then passed through a Time Distributed dense layer where activation is softmax and the model is built .The model is compiled with epoch=50,batch size=128, latent dimension =300, learning rate =0.005,keep probability =0.75 and we used rmsprop optimizer. After the model is built, we analyze a diagnostic plot to understand the model's behaviour over time . Now for prediction , we convert the index to word for source and target vocabulary and build the dictionary . Next, we define the inference function for the encoder and decoder. In the function for the implementation of the inference process, we used two approaches to find the index of the token 1)argmax 2)beam search decoder. Finally, we create the function to change the target sequence from an integer to a word for both reviews and summaries. Few summaries generated by the model as shown in Figure 6.

figure 6. some summaries generated by the model

5. Analysis

We have used ROUGE scores to evaluate the models .ROUGE score is a set of matrices which is used for evaluating text summarization and machine translation in Natural Language Processing. ROUGE-1 and ROUGE-2 comes under ROUGE-N ,which evaluates machine predicted or produced summary against a reference summary by the Overlap of N-grams .Now , ROUGE-1 is the overlapping of unigram(each word) between the predicted summaries and dataset summaries. It is based on LCS(Longest Common Subsequence) statistics. LCS identifies the largest appearing sequence in n-grams by taking into account sentence level structure similarity. ROUGE-2 is the overlapping of bigrams between the predicted summaries and reference summaries.

Based on the scores, we compared 3 models which are given in table 1

table 1: evaluation and comparison of the three models

Model	Evaluation Metrics	F-measure	Precision	Recall
LSTM with Attention Mechanism	ROUGE-1	0.0946889	0.08606516	0.1222280
	ROUGE-2	0.0175438	0.01614035	0.0245614
	ROUGE-l	0.0959580	0.08781954	0.1222280
Bidirectional LSTM with Attention Mechanism	ROUGE-1	0.1887851	0.18487301	0.2423333
	ROUGE-2	0.0161904	0.01366666	0.0258333
	ROUGE-l	0.1869523	0.18403968	0.2373333
Bidirectional LSTM with Attention Mechanism and Word2Vec Embedding Layer	ROUGE-1	0.2225100	0.21203571	0.2738571
	ROUGE-2	0.0686507	0.06566666	0.0750000
	ROUGE-l	0.2220569	0.21333333	0.2698571

The third model which is Bidirectional LSTM with Attention Mechanism and Word2Vec Embedding layer has better performance than the other model on ROUGE-1, ROUGE-2 and ROUGE-l score .The model Bidirectional LSTM with Attention Mechanism has better performance than LSTM with attention mechanism .However , the ROUGE-2 precision value of LSTM is greater than Bidirectional LSTM . The reason for it is that ROUGE precision is given by the formula:

$$\frac{number\_of\_overlapping\_words}{total\_words\_in\_system\_summary}$$

Now since, bidirectional lstm takes care of future as well as past inputs so the total words in system summary increases in comparison to lstm and the value becomes less than the lstm. All the models mentioned in table 1 have higher ROUGE-1 score than ROUGE-2 score .Hence, we can state that the models are effective in generating words correctly for summarization as an individual, but not if taken in word pairs. It means that the system has low performance in producing words correctly, if it is equal or more than 2 word pairs.

We have also used the BLEU score to evaluate the 3 models . BLEU uses the opposite procedure in comparison to ROUGE to evaluate the model. BLEU evaluates the reference summary against the predicted summary whereas ROUGE evaluates the predicted summary against the reference summary. BLEU Score is effective in evaluating the conversion of one language into another BLEU Score of three models are shown in table 2.

Now since, bidirectional lstm takes care of future as well as past inputs so the total words in system summary increases in comparison to LSTM and the value becomes less than the LSTM. In table 1, all the models are having higher ROUGE-1 score in comparison to ROUGE-2 score. Hence, we can state that the models are effective in generating words correctly for summarization as an individual, but not if taken in word pairs. It means that the system has low performance in producing words correctly, if it is equal or more than 2 word pairs. .

We have also used the BLEU score to evaluate the 3 models. BLEU uses the opposite procedure in comparison to ROUGE to evaluate the model. BLEU evaluates the reference summary against the predicted summary whereas ROUGE evaluates the predicted summary against the reference summary. BLEU Score is effective in evaluating the conversion of one language into another. BLEU Score of three models are shown in table 2.

table 2: evaluating bleu score of the three models

Model	BLEU SCORE
LSTM with Attention Mechanism	0.3959138
Bidirectional LSTM with Attention Mechanism	0.3652759
Bidirectional LSTM with Attention Mechanism and Word2Vec Embedding Layer	0.3513953

The results are opposite of that we got for ROUGE scores .In BLEU score, the model lstm with Attention Mechanism has better scores than other models .The model bidirectional LSTM with Attention Mechanism has better performance than the one with embedding layer as Word2Vec .This happened because , BLEU evaluates reference summary with the predicted summary so for the case of bidirectional LSTM the predicted summary contains less word from the reference summary in comparison with the LSTM. In other words bidirectional lstm is able to get the context better and accordingly predicts the summary .So, for most cases

the meaning of the predicted as well as reference summary are same but they contain different words of same meaning and that is why BLEU score is less for Bidirectional LSTM . Pointer generator mechanism resolves this issue, as it is a combination of both abstractive as well as extractive mechanism .Thus, it uses the word present in the reference summary also in the predicted summary along with the context to improve the ROUGE scores and BLEU scores and thus enhances the system's performance.

Finally , we improved the performance of the model bidirectional LSTM with Attention Mechanism and Word2Vec Embedding Layer by using Beam Search Decoder instead of Argument max that is Greedy search algorithm during the implementation of the inference process. Beam search is useful for selecting the best and the most likely word for the target sequence .The beam search algorithm uses conditional probability to choose multiple alternatives from an input sequence at each timestamp [6] whereas Greedy Search algorithm selects one optimized candidate as an input sequence for each timestamp [6].

6. Conclusion and future work

This paper has presented an approach to train the dataset and compare the accuracy of three models using BLEU and ROUGE score. We have also stated the difference between the performance of ROUGE and BLEU score. In the ROUGE score, bidirectional LSTM model performs better than LSTM model and bidirectional LSTM with the Word2vec embedding layer performs better than other two models while in the BLEU score the scenario is opposite since BLEU evaluates reference summary with predicted summary . Finally, we have increased the accuracy in ROUGE score by including beam search decoder in Bidirectional LSTM with attention mechanism and Word2Vec layer.
We will build the model by increasing the training dataset size .In this project we used only 10,000 reviews out of 500,000. The prediction capability of deep neural network models improves with increase in the training dataset. We will implement pointer-generator networks and coverage mechanisms to enhance accuracy and get better results. Pointer generator network is a combination of Abstractive and Extractive mechanism and thus brings in the best from both of them.

References

[1]Dzmitry Bahdanau, K.Cho, Y.Bengio. "Neural Machine Translation by Jointly Learning to Align and Translate". International Conference on Learning Representation (ICLR), (2014).
[2]I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," pp. 1–9(2014).
[3] Puruso Muhammad Hanunggul, Suyanto Suyanto. "The Impact of Local Attention in LSTM for Abstractive Text Summarization" International on Research of Information technology and Intelligent Systems(ISRITI)(2019)
[4] Abu Kaisar Mohammad Masum, Sheikh Abujar, Md Ashraful Islam Talukder, AKM Shahariar Azad Rabby, Syed Akhter Hossain. "Abstractive method of text summarization with sequence to sequence RNNs" International Conference on Computing, Communication, and Networking Technologies(ICCNT)(2019)
[5] Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Ça˘glar G˙ulçehre, Bing Xiang. "Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond"arXiv:1602.06023v5 [cs.CL] (2016).
[6]RanuKhandelwal,"An intuitive explanation of Beam Search",Feb 2,
https://towardsdatascience.com/an-intuitive-explanation-of-beam-search-9b1d744e7a0f
[7]Colah'sBlog,"Understanding LSTM Networks",August 27,2015,
https://colah.github.io/posts/2015-08-Understanding-LSTMs/
[8] JasonBrownie,"A Gentle Introduction to Long Short-Term Memory Networks by the Experts",May 24,2017,
https://machinelearningmastery.com/gentle-introduction-long-short-term-memory-networks-experts/
[9] Synced,"A Brief Overview of Attention Mechanism ",Sep 26,2017,
https://medium.com/syncedreview/a-brief-overview-of-attention-mechanism-13c578ba9129
[10] Dr.Michael J.Garbade ,"A Quick Introduction to Text Summarization in Machine Learning ",Sep 19,2018.
https://towardsdatascience.com/a-quick-introduction-to-text-summarization-in-machine-learning-3d27ccf18a9f
[11] Luis Gonsalves ,"Automatic Text Summarization with Machine Learning - An Overview ",Apr 12,
https://medium.com/luisfredgs/automatic-text-summarization-with-machine-learning-an-overview-68ded5717a25
[12] Arvind Pai ,"Comprehensive Guide to Text Summarization using Deep Learning in Python ",June 10,2019
https://www.analyticsvidhya.com/blog/2019/06/comprehensive-guide-text-summarization-using-deep-learning-python/
[13] Stanford Network Analysis Project ,"Amazon Fine Food Reviews", 2013
https://www.kaggle.com/snap/amazon-fine-food-reviews.