Journal of Current Research in Engineering and Science
JCRES
Bi-Annual Online Journal (ISSN: 2581 - 611X)

VOLUME:8, ISSUE:1, FEBRUARY 2025, PAPER NUMBER:53

Text Summarization using Neural Network Algorithms – BERT and BiLSTM model.

Devanshu Tiwari
Department of Data Science and Business Systems
SRM Institute of Science and Technology
Kattankulathur, Tamil Nadu
dt4590@srmist.edu.in

Mohammed Zaid Ali Syed
Department of Data Science and Business Systems
SRM Institute of Science and Technology
Kattankulathur, Tamil Nadu
ms1756@srmist.edu.in

DR. SHOBANA J.
Department of Data Science and Business Systems
SRM Institute of Science and Technology
Kattankulathur, Tamil Nadu
shobanaj1@srmist.edu.in

Abstract - In today's fast-paced world, the volume of information available can be overwhelming. Text summarization plays a crucial role in various real-life scenarios, such as news aggregation platforms, where users can quickly grasp the essence of multiple articles without reading them in their entirety. This study investigates the use of BiLSTM (Bidirectional Long Short-Term Memory) as well as BERT (Bidirectional Encoder Representations from Transformers) modules in the implementation of a text summarization technique. The algorithm leverages BERT's contextual embeddings to determine the text's semantic meaning and BiLSTM's sequential modelling capability to generate informative summaries. Experiments were conducted using a text document dataset which demonstrated that, in comparison to current methods, the proposed approach produced a more accurate summarization. The evaluation metric ROUGE score is then used to evaluate how well the summary report performed.

Keywords - Text Summarization, BERT, BiLSTM, ROUGE metric.

I. Introduction

The exponential expansion of digital information in recent years has led to an overwhelming influx of textual data in a variety of fields, including social media, academic papers, and reports on the news. Such surge in data underscores the critical importance of automated text summarization techniques, which aim to distill essential information from voluminous documents efficiently.

Traditional methods of text summarization predominantly relied on handcrafted features and shallow learning models, often failing to capture the intricate nuances of natural language. But the discipline of natural language processing (NLP) has undergone a revolution because to the advent of deep learning, especially with sophisticated architectures like BiLSTM (Bidirectional Long Short-Term Memory) and BERT (Bidirectional Encoder Representations from Transformers) [1]. These models offer unparalleled capabilities in understanding and representing textual data with remarkable sophistication.

In this research endeavor, we propose an innovative approach to text summarization by integrating the BERT and BiLSTM architectures.

Pre-trained transformer-based models such as BERT are excellent at extracting semantic meaning and contextual information from text; sequential modeling features of BiLSTM enable it to extract temporal dynamics and long-range dependencies from text. By synergistically harnessing the strengths of these two architectures, we aim to develop a more effective and accurate text summarization framework.

The proposed BERT-BiLSTM text summarization model operates through several key stages. Initially, the input text undergoes tokenization and encoding using the BERT tokenizer, resulting in contextual embeddings for each token. Subsequently, these embeddings are fed into a BiLSTM layer, which processes the sequential information and captures dependencies between tokens. The relevance ratings for each token are then predicted using a linear layer with sigmoid activation, and these scores are used to create the final summary [2].

We benchmark the performance of our BERT-BiLSTM model against traditional summarization techniques and state of the art i.e. SOTA deep learning models. The quality and coherence of the generated summaries are assessed using the evaluation metric known as ROUGE (Recall-Oriented Understudy for Gisting Evaluation) score.

II. Related Work

Related research in the field of text summarization encompasses a broad spectrum of methodologies, spanning from traditional techniques to contemporary deep learning-based approaches [3]. Traditional methods often relied on rule-based systems, statistical algorithms, or graph-based models to extract essential information from textual data. These methodologies typically employed heuristic rules or statistical metrics to identify crucial sentences or phrases for summarization purposes. However, despite their utility, these conventional approaches often struggled to thoroughly convey the text's complex contextual meanings and semantic subtleties, resulting in suboptimal summarization outputs.

Numerous studies have explored the integration of BERT into text summarization frameworks with the objective of leveraging its contextual embeddings to enhance summarization performance. One prevalent approach involves fine-tuning BERT on summarization-specific tasks, where the model learns to generate summaries directly from input text.

Additionally, researchers have investigated the amalgamation of BERT with various different neural network architectures, including LSTM (Long Short - Term Memory) [4] and attention mechanisms, to further augment summarization performance. These hybrid models harness the strengths of each architecture, with BERT capturing contextual information and the additional components modeling sequential dependencies and attention mechanisms to enhance summarization quality.

While BERT-based methodologies have exhibited promising results in text summarization, ongoing research endeavors continue to explore novel architectures and training strategies to advance the current state of the art or the present status quo. Overall, there is a lot of promise for improving the capabilities of automated summarizing systems and enabling effective information extraction from textual data through the integration of BERT and deep learning techniques into text summarization.

III. Proposed Architecture

In the proposed architecture for text summarization leveraging BERT-BiLSTM, several key components and methodologies contribute to its effectiveness and performance. Here are some points to highlight in the research paper:

Integration of BERT and BiLSTM: The architecture encompasses the advantages of BiLSTM [5], which is excellent at representing sequential dependencies, with the capabilities of BERT, a transformer-based model well known for its capacity to collect contextual information. By integrating these two architectures, this approach can efficiently extract the local as well as global semantic information from the supplied text document.

Tokenization and Encoding: The input text undergoes tokenization and encoding using the BERT tokenizer [6], generating contextual embeddings for each token.

BiLSTM Layer: The contextual embeddings from BERT are then fed into a bidirectional LSTM (BiLSTM) layer, which processes the sequential information and captures dependencies between tokens. When producing summaries, the bidirectional LSTM enables the model to consider both past and future context, giving outputs that are more robust and contextually relevant.

Scoring Mechanism: After processing through the BiLSTM layer, a linear layer with sigmoid activation is employed to anticipate the importance review scores for every token in the text document given as input.

Summary Generation: The top-ranked tokens in the text document are chosen to produce the summary after the tokens have been rated according to their importance scores [7].

Fine-tuning and Optimization: To adapt its representations to the summarization domain, the pre-trained BERT model will have to be fine-tuned on summarization-specific tasks in the proposed architecture[8].

Evaluation Metrics: A common evaluation metric, such as the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) score, is used to assess the performance of the suggested architecture.

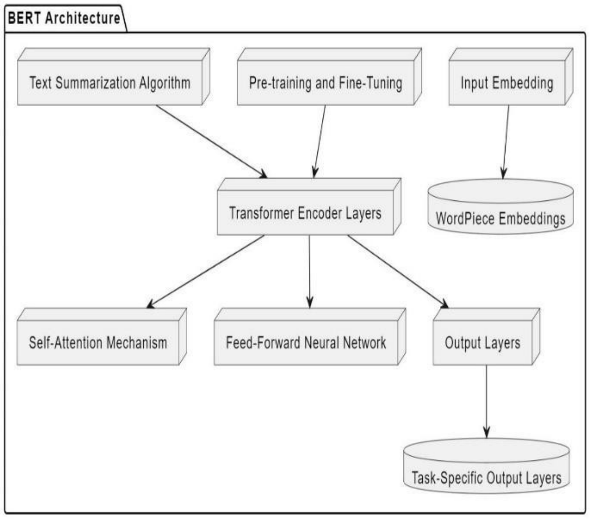
BERT Architecture
Text Summarization Algorithm
Pre-training and Fine-Tuning
Input Embedding
Transformer Encoder Layers
WordPiece Embeddings
Self-Attention Mechanism
Feed-Forward Neural Network
Output Layers
Task-Specific Output Layers

Figure1. Structure of the proposed BERT architecture with both BERT Transformers.

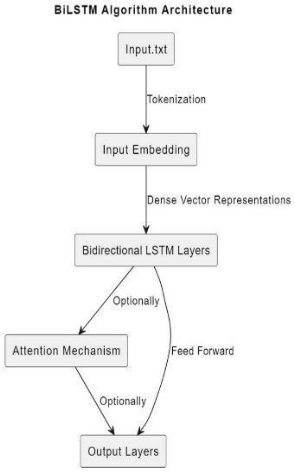

Figure2. The structure of the proposed BiLSTM architecture

A recurrent neural network (RNN) architecture known as a BiLSTM network makes predictions at each time step by combining data from both previous and future states. It has two layers of LSTMs, one processing the input sequence forward and the other processing it backward. Concatenating the outputs from these two layers gives the model data from contexts in the past and future [9].

The input sequence is represented by the notation *X*=(*x1,x2,...,xT*)*,* where T is the sequence's length. The BiLSTM architecture consists of the following components:

Input Layer: The input sequence X is inserted into the network.

Embedding Layer: Transforms input tokens into continuous vector representations via the embedding layer. If used, the embedding layer transforms the input sequence into *E*=(*e*1,*e*2,...,*eT*), where *et* is the embedding vector for the token at time step *t*.

Forward LSTM Layer: The input sequence is processed in forward order by the forward LSTM layer. The following formulas are used by the forward LSTM to calculate hidden states, or *htf*, at each time step *t*:

$$i_t^f = \sigma(W_{ix}^f x_t + W_{ih}^f h_{t-1}^f + b_i^f)$$

$$f_t^f = \sigma(W_{fx}^f x_t + W_{fh}^f h_{t-1}^f + b_f^f)$$

$$o_t^f = \sigma(W_{ox}^f x_t + W_{oh}^f h_{t-1}^f + b_o^f)$$

$$\tilde{c}_t^f = \tanh(W_{cx}^f x_t + W_{ch}^f h_{t-1}^f + b_c^f)$$

$$c_t^f = f_t^f \odot c_{t-1}^f + i_t^f \odot \tilde{c}_t^f$$

$$h_t^f = o_t^f \odot \tanh(c_t^f)$$

Backward LSTM Layer: The input sequence is processed in reverse order by the Backward LSTM Layer. It computes hidden states (*htb*) at each time step *t* in a manner akin to the forward LSTM, except in reverse order.

Concatenation Layer: The final output sequence is created by concatenating the forward and backward hidden states at every time step. The concatenation of forward and backward hidden states is represented by the notation, *H*=(*h*1,*h*2,...,*hT*), *ht*=[*htf*,*htb*].

Output Layer: Depending on the task, the output layer could vary. In sequence labeling tasks like as named entity recognition or part-of-speech tagging, a SoftMax layer is usually used to predict the label for every token in the sequence.

Once trained, the BiLSTM network can be evaluated on a separate dataset using metrics appropriate for the specific task, such as accuracy, F1 score, or BLEU score for machine translation.

Transformer Encoder: A stack of transformer encoder layers makes up BERT. Each layer consists of multi-head self-attention processes and Position-wise feed-forward neural networks. A series of contextualized representations, *H*=(*h*1,*h*2 ,...,*hT*) are generated by the encoder from a sequence of input tokens *X*=(*x*1,*x*2,...,*xT*) and produces a sequence of [10].

Pre-training Tasks: Two unsupervised tasks are used to pre-train BERT.

a. Masked Language Modeling (MLM) involves the random masking of certain tokens inside the input sequence, with the model being trained to predict the masked tokens according to their context [11].

b. Next Sentence Prediction (NSP): The model learns to predict whether the second sentence in an original textual pair follows the first one.

Fine-tuning: By adding task-specific layers to the pre-trained BERT model and training on task-specific data, BERT can be improved on downstream tasks following pre-training [12].

Let's denote:
Encoder(X): Function representing the encoder, which produces the contextualized representations H.

MaskedLM(X): The function MaskedLM(X) predicts masked tokens in the input sequence and represents the masked language modeling job.

NSP(S1,S2) : Function that indicates whether the second sentence (S2) will come after the first phrase (S1) in the next sentence prediction task [13].

Here is a summary of the mathematical formulation for BERT:

$$H = \text{Encoder}(X)$$

$$P(x_i|X_{\text{masked}}) = \text{MaskedLM}(X)$$

Given pairs of sentences (S_1, S_2), BERT predicts whether S_2 follows S_1:

$$P(\text{IsNext}|S_1, S_2) = \text{NSP}(S_1, S_2)$$

Large text corpora are used to pre-train BERT, and task-specific data are used to refine it. Following fine-tuning, task-specific evaluation metrics like accuracy, F1 score, or ROUGE score are used to assess BERT on a variety of downstream tasks such text categorization, named entity identification, and question answering [14].

IV. Experiment and Analysis

Experiment Setup - In the experiment setup for the research paper, several crucial steps were undertaken. Firstly, appropriate datasets were selected for training and evaluation. The datasets were then pre-processed (tokenization, padding, and partitioning into training, validation, and test sets) to convert unprocessed text into a format appropriate for model training.

The training procedure was then meticulously executed, employing appropriate optimizers, learning rates, batch sizes, and epochs, while continuously validating against a validation set to mitigate overfitting risks. Evaluation metric such as ROUGE were chosen to gauge model performance, supplemented by potential human evaluation for qualitative assessment. The experiments were conducted systematically, with attention to reproducibility by documenting all hyperparameters, random seeds, and relevant details, and results were averaged across multiple runs to ensure reliability. Through this comprehensive experiment setup, researchers aimed to systematically evaluate the efficacy of BERT and BiLSTM models for text summarization tasks.

Evaluation Metric - The suggested model's performance is assessed by the application of the ROUGE assessment metric. The ROUGE-1 score is calculated by identifying the unigram overlap between summaries generated by the system and summaries generated by humans. The computation of the ROUGE-2 score involves identifying the overlap of bigrams between summaries generated by the system and those generated by humans. Rouge-L gauges the lengthiest typical sequences. Table 1 presents the tabulated findings of three evaluation measures.

Baseline summarization models - RNN, Seq-Seq model with attention, and Pointer generator + attention mechanism are the three base-line summarization techniques that are most frequently utilised. The outcomes derived from these models are contrasted with those of the suggested model. These models' brief descriptions are as follows:

BERT**:** Google created the well-known deep learning model for natural language processing called BERT. It collects contextual data in both directions by using a transformer architecture [15].

Seq2Seq**:** Represents a baseline attention-based standard seq2seq neural network model, which is used to enhance the text's semantics in summaries.

BERT + BiLSTM**:** This model combines BERT, a powerful contextual language model, with a bidirectional LSTM (BiLSTM) network to generate abstractive summaries. BERT provides pre-trained contextual embeddings capturing rich semantic information, while the BiLSTM captures sequential dependencies in the input text.

V. Results

The findings of the suggested model and other popular baseline models are tallied in Table 1 of this section. The comparison between the proposed model and existing classical model are analyzed based on input document dataset.

Table 1. Outcomes of the baseline summarization models and the proposed model on the input document dataset

Summarization Model	**ROUGE -1**	**ROUGE -2**	**ROUGE -L**
Seq2Seq	29.4	22.5	29.4
BERT	39.7	31.6	39.7
Proposed Model (Implementation of BERT with Bidirectional LSTM)	62.5	55.5	62.5

The Rouge-1 scores for the models are depicted in the graph below.

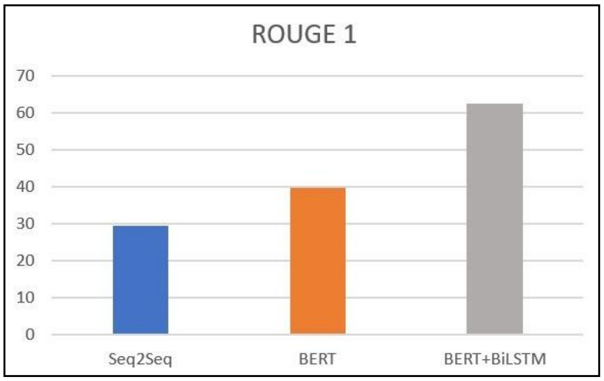

Figure3. Rouge-1 score comparison

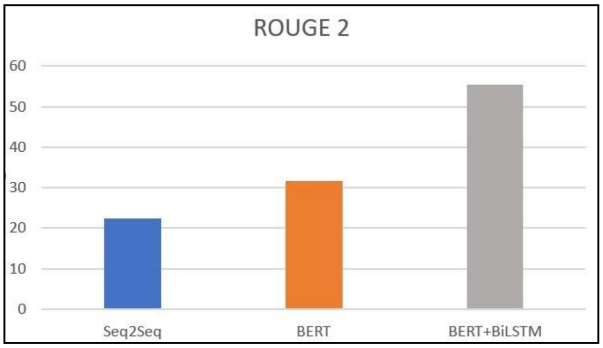

Figure4. Rouge-2 score comparison

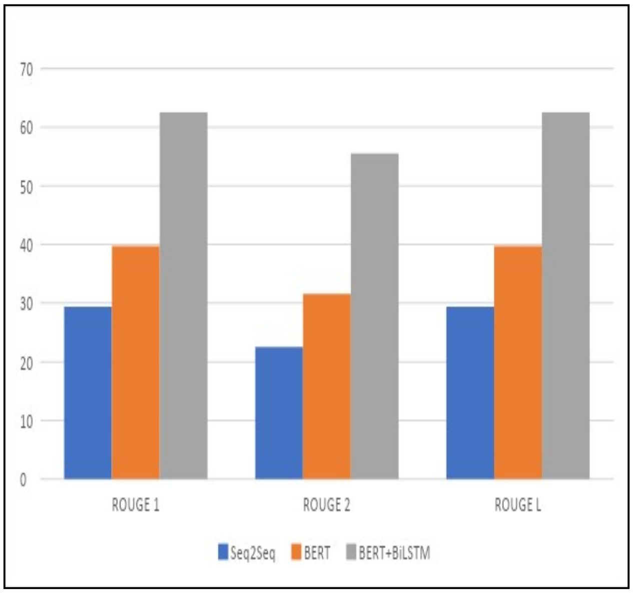

Figure5. Table 1 Comparative Analysis graph for proposed model.

VI. References

[1] Miller, D. (2019). Leveraging BERT for Extractive Text Summarization on Lectures. *ArXiv*. /abs/1906.04165

[2] S, Kavyashree & R, Sumukha & R, Soujanya & V, Tejaswini. (2023). Survey on Automatic Text Summarization using NLP and Deep Learning. 523-527. 10.1109/ICAECIS58353.2023.10170660.

[3] M. S M, R. M P, A. R E and E. S. G SR, "Text Summarization Using Text Frequency Ranking Sentence Prediction," 2020 4th International Conference on Computer, Communication and Signal Processing (ICCCSP), Chennai, India, 2020, pp. 1-5, doi: 10.1109/ICCCSP49186.2020.9315203.

[4] I. Awasthi, K. Gupta, P. S. Bhogal, S. S. Anand and P. K. Soni, "Natural Language Processing (NLP) based Text Summarization - A Survey," 2021 6th International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 2021, pp. 1310-1317, doi: 10.1109/ICICT50816.2021.9358703.

[5] A. Jadhav, R. Jain, S. Fernandes and S. Shaikh, "Text Summarization using Neural Networks," 2019 International Conference on Advances in Computing, Communication and Control (ICAC3), Mumbai, India, 2019, pp. 1-6, doi: 10.1109/ICAC347590.2019.9036739.

[6] Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2019). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. *ArXiv*. /abs/1910.13461

[7] Dong, L., Yang, N., Wang, W., Wei, F., Liu, X., Wang, Y., Gao, J., Zhou, M., & Hon, H. (2019). Unified Language Model Pre-training for Natural Language Understanding and Generation. *ArXiv*. /abs/1905.03197

[8] Zhang, J., Zhao, Y., Saleh, M., & Liu, P. J. (2019). PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. *ArXiv*. /abs/1912.08777

[9] M. Ji, R. Fu, T. Xing and F. Yin, "Research on Text Summarization Generation Based on LSTM and Attention Mechanism," 2021 International Conference on Information Science, Parallel and Distributed Systems (ISPDS), Hangzhou, China, 2021, pp. 214-217, doi: 10.1109/ISPDS54097.2021.00048.

[10] Zhong, M., Liu, P., Chen, Y., Wang, D., Qiu, X., & Huang, X. (2020). Extractive Summarization as Text Matching. *ArXiv*. /abs/2004.08795

[11] Yuan, R., Wang, Z., & Li, W. (2020). Fact-level Extractive Summarization with Hierarchical Graph Mask on BERT. *ArXiv*. /abs/2011.09739

[12] P. Raundale and H. Shekhar, "Analytical study of Text Summarization Techniques," 2021 Asian Conference on Innovation in Technology (ASIANCON), PUNE, India, 2021, pp. 1-4, doi: 10.1109/ASIANCON51346.2021.9544804.

[13] Widyassari, Adhika & Rustad, Supriadi & Shidik, Guruh & Noersasongko, Edi & Syukur, Abdul & , Affandy & Setiadi, De Rosal Ignatius Moses. (2020). Review of Automatic Text Summarization Techniques & Methods. Journal of King Saud University - Computer and Information Sciences. 34. 10.1016/j.jksuci.2020.05.006.

[14] Shi, Kaile et al. “StarSum: A Star Architecture Based Model for Extractive Summarization.” IEEE/ACM Transactions on Audio, Speech, and Language Processing 30 (2022): 3020-3031.

[15] Liu, Y. (2019). Fine-tune BERT for Extractive Summarization.