Improving the BERT model for long text sequences in question answering domain

Vijayan Ramaraj, Mareeswari Venkatachala Appa Swamy, Ephzibah Evan Prince, Chandan Kumar
Department of Software and Systems Engineering, School of Computer Science Engineering and Information Systems (SCORE), Vellore Institute of Technology (VIT), Vellore, India

Article Info

Article history:

Received Aug 18, 2023
Revised Oct 15, 2023
Accepted Nov 28, 2023

Keywords:

BERT
Chatbot
Long text processing
Natural language processing
Question-answering system
SQuAD
Transformer

ABSTRACT

The text-based question-answering (QA) system aims to answer natural language questions by querying the external knowledge base. It can be applied to real-world systems like medical documents, research papers, and crime-related documents. Using this system, users don't have to go through the documents manually the system will understand the knowledge base and find the answer based on the text and question given to the system. Earlier state-of-the-art natural language processing (NLP) was recurrent neural network (RNN) and long short-term memory (LSTM). As a result, these models are hard to parallelize and poor at retaining contextual relationships across long text inputs. Today, bidirectional encoder representations from transformers (BERT) are the contemporary algorithm for NLP. BERT is not capable of handling long text sequences; it can handle 512 tokens at a time which makes it difficult for long context. Smooth inverse frequency (SIF) and the BERT model will be incorporated together to solve this challenge. BERT trained on the Stanford question answering dataset (SQuAD) and SIF model demonstrates robustness and effectiveness on long text sequences from different domains. Experimental results suggest that the proposed approach is a promising solution for QA on long text sequences.

Corresponding Author:

Mareeswari Venkatachala Appa Swamy
Department of Software and Systems Engineering, School of Computer Science Engineering and Information Systems (SCORE), Vellore Institute of Technology (VIT)
Vellore, Tamil Nadu 632014, India
Email: vmareeswari@vit.ac.in

1. INTRODUCTION

Terabytes of data are generated every day. Information is the new currency in today's world. Extracting knowledge from the data is a tedious task, so there is a need to build a system that can extract information/knowledge from the given user input for different applications such as COVID-19 [1] and agriculture [2]. So, a question-answering system (QA) can be developed that can take user queries as well as a piece of text i.e., a paragraph in natural language and can provide relevant answers from the given paragraph.

One of the key advantages of a QA system is that it can help users save time and effort when searching for information, by quickly and accurately providing them with relevant answers. This makes it particularly useful for applications such as emergency operations [3], customer service, education, and research [4], where users need to access and analyze large volumes of text data to find the information they need. QA system has a wide application in various fields. It can be used in the fields of medicine, crime, education, and telecommunication. Building a QA system requires a large corpus. An excellent application of the QA system will be in the medical industry where patient health record history is feed to the system and doctors/expert can ask relevant question from the system and in this doctor don't have to read patient medical record. Another application can be in police investigation, where all the records of the culprit and victim and feed to system and police can ask question from the system and it will save a lot of time as the police do not have to read or go through the crime files. Many QA models have been developed till date.

The bidirectional encoder representations from transformers (BERT) [5] based QA system for long text has a lot of potential applications, including in fields such as education, research, and customer service. It was developed by Google. By providing users with accurate and relevant answers to their queries, this system has the potential to streamline information access, improve productivity, and enhance the overall user experience. Overall, a QA system for long text is an advanced and sophisticated technology that might completely change how users consume information. By giving accurate and relevant answers to their queries, these systems can help to streamline research, improve productivity, and enhance the overall user experience. These authors [6] developed the QA system COBERT for answering the queries by searching the 59,000 COVID-19-associated articles. They used the BERT model on the Stanford question answering dataset (SQuAD) 1.1 dev dataset, generated the ranks of search results, and finalized the answer with a small description, article title, and source of the article.

Hierarchical models like the hierarchical attention network [7] or the transformer-XL are used to process long text sequences. These models are effective for tasks like document classification and text generation [8]. Many transformer models do not process more than 512 tokens, hence multi-labeling is applied in long documents [9]. Recent research has also been explored in adapting BERT to handle long text sequences more efficiently [10]. For example, the [11] longformer model extends BERT to handle the writing essays of students and generate feedback about their writing skills. They proved longformer model is better than BERT by providing more attention to the categorization of various human languages. Many transformer-based models have been developed recently like XLNet, sparse transformers, and BigBird. BigBird [12] is a transformer-based model that has been designed for handling long sequences. It uses a combination of random attention, window attention, and global attention to process sequences up to tens of thousands of tokens.

BERT is a complicated model that necessitates a sizable investment in processing resources during the training phase. It can be expensive and take several days to train a BERT model on a sizable dataset using numerous GPUs [13]. BERT is a pre-trained model that has to be fine-tuned on a few particular downstream jobs. BERT needs a lot of task-specific data, which might not always be accessible, to be tuned, though. Small dataset fine-tuning may lead to overfitting and poor generalizability of the model [14]. Lack of ability to handle terms outside of its vocabulary: BERT has a fixed vocabulary size and is unable to handle words outside of its vocabulary. For some languages or for terminology that may not be included in the vocabulary. BERT works based on character level checking but the Chinese language is designed based on phrases, this can be an issue [15]. To improve the accuracy and speed retrieval system, they [16] increase iteration to optimize the BERT model for processing huge Japanese sentences.

Some of the existing research that used the BERT model for QA, such as SQuAD [17], natural questions, BioASQ, and DuoRC. BERT has been used as a popular SQuAD benchmark for QA and achieved state-of-the-art performance on SQuAD 1.1 and SQuAD 2.0, TriviaQA[18], outperforming previous models. BERT has also been used for the natural questions dataset, which is a more challenging QA dataset than SQuAD. The model reached a high performance on the task of long-form QA. BERT has been used for the TriviaQA dataset, which is a more difficult dataset that requires answering questions based on both text passages and external knowledge sources. BERT has been used for the biomedical QA task in the BioASQ challenge. The model achieved state-of-the-art performance on this task, outperforming previous models. BERT has also been used for the DuoRC dataset, which is a new dataset for QA that requires reasoning about multiple sentences in a paragraph. BERT can be practiced with graph neural networks to handle unstructured data like text and images [19].

QA models have high memory requirements. Shim *et al.* [20] propose a solution for reducing excessive memory usage. When this method is used with any recurrent neural network (RNN) model combined with an input encoder, context reduction layer, and context attention layers it helps in reducing training time and memory consumption. Khin and Soe [21] proposed a framework for natural language processing (NLP) QA systems using long short-term memory (LSTM)-RNNs. But still, RNN models are lacking in high dimensional data representation and biased attentive weight assignment. Resolving this issue by Bi-directional LSTM (BiLSTM) with an attentive mechanism is proposed [22]. Another approach used by Hanifah and Kusumaningrum [23] for improving user intention understanding in QA systems using attention-based LSTM networks. Enamoto *et al.* [24] trained the model with one BiLSTM layer and one attention layer over on short length of legal documents. The gradient problem in RNN has been resolved by a simple recurrent unit (SRU) based self-matching network (S^2-Net), it achieved an 80.8% FI score on the SQuAD dataset [25].

The authors note that traditional keyword-based approaches to NLP-QA systems are limited by their inability to understand the nuances of human language, and argue that deep learning techniques such as LSTM-RNNs can improve performance by allowing the system to learn from contextual information. Even these supervised methods perform less than the smooth inverse frequency (SIF) model [26]. The SIF method performs well on the textual similarity tasks by unweighted average. The SIF model is an adaptable method for all domain settings in different test beds to compute the sentence embedding by training word vectors. The mechanism of word frequency analysis and weighted method will improve the performance of the model. Overall, BERT has demonstrated impressive performance on a variety of QA tasks, including both single-sentence and multi-sentence QA, as well as biomedical QA. These results suggest that BERT is a highly effective model for QA and has the potential to be used in a wide range of applications. Even though, the BERT model can be applied only for handling textual inputs with 512 tokens. Hence, this proposed work combines the BERT and SIF methods to process the long sentences of the QA system of any application. The following section deliberates the practiced research methods and discusses the experimental comparisons and insight into the significance.

2. RESEARCH METHOD

QA system helps in extracting information from the given document with user query. Using this kind of system enables users to save time, cost, and effort. SQuAD will be used to train the model. The objective is to handle long sequence input text for QA as BERT is limited to 512 tokens input. QA system should not have a significant difference in accuracy at the same time. The proposed architecture as shown in Figure 1 is based on the BERT model and SIF. It is a transformer-based attention model, which is used for NLP tasks like QA, sentiment analysis, and inference analysis. The proposed model is capable of handling longer text sequences for QA tasks.

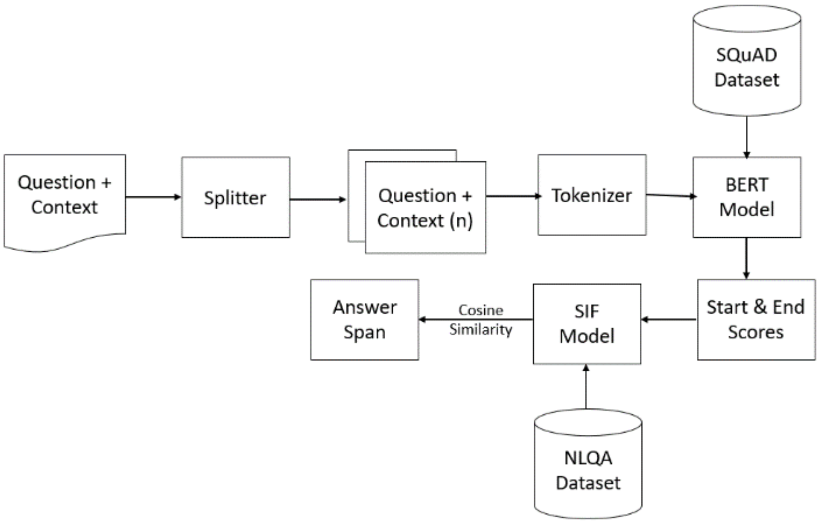


Figure 1. System architecture

The basic idea behind the proposed model is as follows. Here BERT-base-uncased-SQuAD model will be used which is already trained on the SQuAD V2 dataset. The model has 12 repeating layers, 128 embedding, 4096-hidden, 64-heads, and 223M parameters. First BERT will be trained on the SQuAD dataset then the SIF embedding model will be trained on the glove-wiki-gigaword-100 dataset, and it will be initialized first. This dataset acts as a pre-trained foundation because the context articles in the dataset are based on Wikipedia articles. This will help in enhancing the basic embeddings since the model's learning was limited by the training set's 4,000 data points. After the model has been taught. The question's SIF encoding is first identified. The answer is then anticipated to come from the substance of the corresponding piece, which will be broken down into individual phrases. Questions and context will be passed to the splitter module. The context or article has 3,000 words to 12,000 words. It will create context chunks of size 250 tokens. The first chunk will have 250 tokens and the next chunks will have the previous 100 tokens as well as the new 150 tokens and so on. Earlier, when a token size of 512 was used the answer was not complete because the answer was getting split, since the answer can be in two passages. So that is why the above method has been used. Now, these chunks will be passed to the input layer. The input layer takes in the question and passage as text inputs. These inputs are tokenized and encoded using a pre-trained BERT model. The BERT encoder is a stack of transformer layers that encode the question and passage into a set of contextualized word embeddings. The embeddings capture the meaning and context of each word in the input. Now, chunks will be passed to the BERT model one by one and for each chunk, BERT will give start and end scores. These start and end scores refer to the starting and ending indices of the answer. Now, the answer text will be extracted between these indices. Now, these answer texts will be fed to the SIF model. SIF model uses pre-trained glove embedding i.e., glove-wiki-gigaword-100 and on top of that natural language QA dataset will be trained. The output of BERT with questions will be fed into this model. For each answer similarity score will be produced as output. The output with the highest similarity will be chosen as the final answer.

2.1. Input representation

The first module of the proposed work is a representation of inputs to the BERT model. This representation includes the summation of token embedding, segmentation embedding, and position embedding. Generally, the embedding method is used in the transcription of the words into number formats in NLP models. BERT uses WordPiece embedding input for tokens. To create word-piece embeddings, each word in the text is first broken up into many smaller words, or "word pieces." The method used for this is known as byte-pair encoding, which creates a collection of word pieces that can be used to symbolize any word in the corpus of text by repeatedly joining the most common character pairs. Each word fragment from the text is transferred to a fixed-size vector after it has been divided into word fragments. In the case of the BERT model, the token embeddings are generated using a combination of techniques that make use of both the context of the word and its position within the text. In Figure 2, a classification (CLS) token has been added in the beginning. A specific (SEP) token has been added to distinguish between question and context.

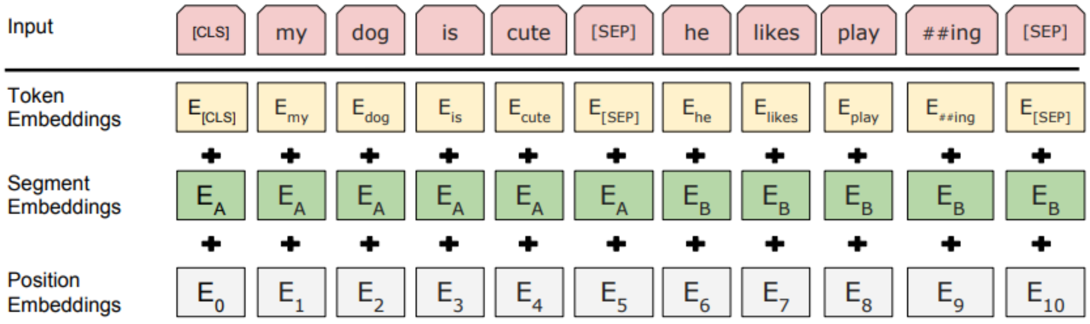


Figure 2. BERT input representation [2]

In NLP, a segment refers to a contiguous sequence of tokens in a text input. For example, in a document that contains multiple paragraphs, each paragraph can be considered a segment. Segment embedding is created by mapping each token in a segment to a fixed-size vector that represents the segment it belongs to. In BERT, the segment embedding is represented as a binary vector of the same size as the token embedding, where each position in the vector corresponds to a token in the input sequence. The value of each position in the segment embedding is set to either 0 or 1, depending on whether the corresponding token belongs to the first segment (e.g., the question) or the second segment (e.g., the document being searched for an answer). Segment embedding is particularly useful in NLP tasks including QA, text classification, and language modeling. Together, these techniques help NLP models like BERT and XLNet to achieve highly effective representations of text that capture both the meaning and structure of the input.

Positional embedding is used in conjunction with other embedding, such as word embedding or character embedding, to provide the model with a representation of the relative positions of the words in the input sequence. In NLP, input sentences are often treated as sequences of tokens (e.g., words or characters), and each token is represented as a high-dimensional vector. However, since these embeddings are fixed and lack information about the order or position of the words, the model may have difficulty distinguishing between words in different positions. To overcome this problem, positional embedding assigns a unique position vector to each token based on its position in the input sequence. The position vectors are added to the token embedding and are learned by the model during training. This provides the model with information about the position of each token in the sequence, which can help it better capture the relationships between words in different positions. There are different types of positional embedding, including sine, and cosine positional encodings, which are commonly used in transformer-based models like BERT and generative pre-trained transformer (GPT). These positional embedding encodes the relative position of each token in the input sequence using sine and cosine functions with different frequencies and phases.

2.2. Splitter

This takes the encoded form of input which has token embedding, segment embedding, and positional embedding. From this, input chunks of 512 tokens will be created. Each chunk will have "CLS token + question tokens + SEP token + context tokens + SEP token". Here, 101 represents the CLS token, and 102 represents the SEP token.

2.3. Attention layer

Despite being simply a weighted mean reduction, an attention layer is motivated by concepts of human attention. The query, the values, and the keys are the three inputs of the attention layers. These inputs are frequently the same, with the query being one key and the keys and values being the same. It excels in modeling a series of events, like language. A soft-max activation function may be used to normalize the attention vector; however, the attention mechanism equation is a hyperparameter. With the help of the attention mechanism, attention layers calculate an attention vector, which is subsequently reduced by computing an attention-weighted average. The determined output of the attention mechanism is returned while using hard attention (hard max function).

2.4. Masked language model

Masked language model (MLM) is designed with the concept that they can learn the relationships between words in context, allowing them to produce more text in natural language. NLP activities such as text classification, named entity recognition, and language translation have all made use of MLMs, BERT is one kind of MLM. BERT's pre-training on massive text corpora enables it to be tailored for particular operations with only a lesser amount of extra training data. A user enters a sentence with some words hidden to use a mask language model. From the perspective of the context of a word, the model forecasts the words that are most likely to fill in the missing blanks. This is beneficial for jobs like text completion, where the model can produce more grammatically correct text. Overall, mask language models have shown to be very effective in a range of NLP tasks, and as the field develops their significance is likely to grow.

To enhance the quality of the word embedding in the model and help it better comprehend the context [27] and a connection between sentences in text, the next sentence prediction is used as a pre-training exercise. The model's goal during pre-training is to forecast whether the second phrase in a pair will follow the first sentence in the text or not. To determine whether the second sentence in the text follows the first sentence or not, the contextualized embedding for the [CLS] token, which is the first token of the first sentence, is then passed through a fully connected layer and a soft-max function. Using the cross-entropy loss function, the model is pre-trained to maximize this likelihood value. The model can acquire crucial language characteristics like coherence and causality in the connections between various sentences in a text by utilizing next-sentence prediction in pre-training BERT. The ability of the model to integrate both local and global background information in its depictions of text is a significant benefit of using next-sentence prediction in pre-training.

2.5. Smooth inverse frequency model

SIF is a technique used in NLP to generate word embedding that can be used to represent the meaning of words in a text. The goal of SIF is to create word embedding that is less sensitive to the rate of occurrence of each word in the text, while still capturing its meaning. The basic idea behind SIF is to first generate a weighted average of the word embedding in the text, where the weights are inversely proportional to the frequency of occurrence of each word. This is done to give more weight to words that occur less frequently in the text since these words are likely to be more informative about the meaning of the text. To further advance the quality of the resulting word embedding, a technique called "common component removal" is applied. This technique involves subtracting from each word and embedding its projection onto a set of common components that are shared by all words in the text. The common components are obtained by taking the singular value decomposition (SVD) of the word embedding matrix and selecting only the top few singular vectors. By using the SIF technique to generate word embedding, it is possible to obtain embedding that is more robust to the frequency of occurrence of each word in the text, while still capturing its meaning. These embeddings can then be used as input to a wide range of NLP tasks, such as text classification, clustering, and information retrieval.

2.6. Dataset

The dataset is the question-answer dataset by Rachael Tatman, which includes the following characteristics: ArticleTitle is the title of the Wikipedia article that served as the source for the initial inquiries and responses; the question; the answer; DifficultyFromQuestioner is the suggested difficulty level that was provided to the question-writer; DifficultyFromAnswerer is an evaluation of the question's difficulty given by the person who evaluated and responded to it; it may be different from the difficulty in field 4; the file containing the particular article is called ArticleFile.

Some of the data points have NAN, empty values in the ArticleFile, and answers columns, so those data points have been dropped. The article text is in a different text file and the question, and answer are in a different file. So, for each data point text article has been extracted from the article file. DificultyFromQuestioner. DificultyFromAnswerer and ArticleFile are dropped from the data points since it was not needed. The text in ArticleText, question, and answer has been cleaned using regex and converted to lower-case characters. The baseline code has been modified and added splitter module for splitting the context. The SIF model is augmented to the BERT model as well. The machine used has 12 GB of RAM. Xeon Processors @2.3 GHz have been used. Python language, PyTorch, transformers, numpy, and pandas libraries were used for implementation.

3. RESULTS AND DISCUSSION

Compare the result obtained with other QA systems which are based on SQuAD, Natural Language, and CoQA could not be done because SQuAD has tokens less than 512, CoQA is a conversational dataset and NL dataset has more than 512 tokens but it has answers directly from the context. The dataset that is being used has different types of answers, so the result cannot be compared on these datasets. There is no existing work found that is built using the dataset which is used.

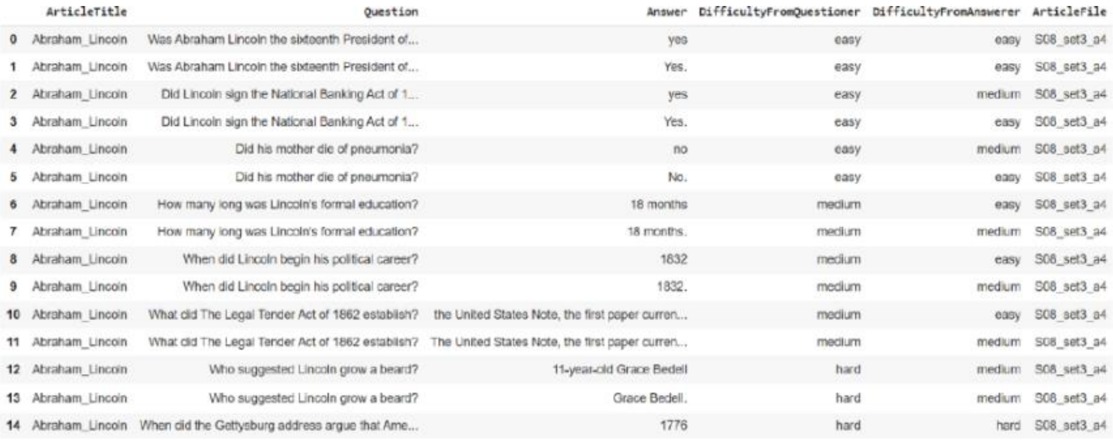
Here dataset is being described which has approximately 4,000 rows and 6 columns such as ArticleTitle, question, answer, DifficultyFromQuestioner, DifficultyFromAnswer, and ArticleFile, see Figure 3. The important columns for the project are article text, Question, and Answer. The article text is the context which has words from 500 to 12,000 words, refer to Figure 4. Figure 5 shows the number of words available in each question. The number of words varies in question from 3 to 26 words approximately. Most of the questions have 8 to 12 words. Some answers are directly picked from the context or article text, some answers are yes and no while some answers were written by users themselves.

Figure 5 shows some output produced by the proposed model. In the first example, the real answer that is available in the answer column dataset is mimicry but the proposed model gave camouflage as an answer because in the article column of the dataset, the mimicry word is not used and the model finds the answer from the article text. Mimicry and camouflage have the same meaning. In the second example, the answer is the same but some of the words of the answer are divided because BERT converts the words to its unit called the token. For the third example predicted answer is incorrect. The sample test cases can be referred to in Figure 6 and Table 1.

Compare the result obtained with other QA systems which are based on SQuAD, Natural Language, and CoQA could not be done because SQuAD has tokens less than 512, CoQA is a conversational dataset and NL dataset has more than 512 tokens but it has answers directly from the context. The dataset that is being used has different types of answers, so the result cannot be compared on these datasets. There is no existing work found that is built using the dataset which is used. Accuracy and F1 score could not calculated directly but a workaround to this problem is used as follows. Computation of cosine similarity between actual answers and outputted answers from the model was done for 400 random samples. In the QA task, the F1 score and exact match (EM) score are considered. Table 2 shows the performance comparison between different models which are the existing vanilla BERT model, the BERT model trained on the SQuAD dataset, and the BERT+SIF model. The proposed BERT+uSIF model produces roughly 80% accuracy with an EM score of 54.28% and an F1 score of 85.53%.

The uSIF model is better than the Word2Vec model because now embeddings are sentence-based and not word-based. Sentence Embedding can only tell which line or sentence contains the answer but is not able to find the exact answer from the context. The BERT base model trained on the SQuAD dataset has better performance than the BERT base model. The BERT+uSIF model outperformed both the BERT base and BERT (SQuAD) in terms of accuracy but has a lower EM score. Training the BERT on the SQuAD dataset and training the SIF model on Natural Language QA has increased the accuracy of the model.

Now, the model can give pin-point answers to the questions based on the context of any length. It was made possible by incorporating the BERT layer with the SIF model. It gave a lower EM score than BERT base and BERT SQuAD, because the dataset used, has approximately 30% of the answers in the yes and no form, and some answers are not directly picked from the context, which is why it has a lower EM score. Hence, the proposed model produces higher accuracy than other BERT models.



	ArticleTitle	Question	Answer	DifficultyFromQuestioner	DifficultyFromAnswerer	ArticleFile
0	Abraham_Lincoln	Was Abraham Lincoln the sixteenth President of...	yes	easy	easy	S08_set3_a4
1	Abraham_Lincoln	Was Abraham Lincoln the sixteenth President of...	Yes.	easy	easy	S08_set3_a4
2	Abraham_Lincoln	Did Lincoln sign the National Banking Act of 1...	yes	easy	medium	S08_set3_a4
3	Abraham_Lincoln	Did Lincoln sign the National Banking Act of 1...	Yes.	easy	easy	S08_set3_a4
4	Abraham_Lincoln	Did his mother die of pneumonia?	no	easy	medium	S08_set3_a4
5	Abraham_Lincoln	Did his mother die of pneumonia?	No.	easy	easy	S08_set3_a4
6	Abraham_Lincoln	How many long was Lincoln's formal education?	18 months	medium	easy	S08_set3_a4
7	Abraham_Lincoln	How many long was Lincoln's formal education?	18 months.	medium	medium	S08_set3_a4
8	Abraham_Lincoln	When did Lincoln begin his political career?	1832	medium	easy	S08_set3_a4
9	Abraham_Lincoln	When did Lincoln begin his political career?	1832.	medium	medium	S08_set3_a4
10	Abraham_Lincoln	What did The Legal Tender Act of 1862 establish?	the United States Note, the first paper curren...	medium	easy	S08_set3_a4
11	Abraham_Lincoln	What did The Legal Tender Act of 1862 establish?	The United States Note, the first paper curren...	medium	medium	S08_set3_a4
12	Abraham_Lincoln	Who suggested Lincoln grow a beard?	11-year-old Grace Bedell	hard	medium	S08_set3_a4
13	Abraham_Lincoln	Who suggested Lincoln grow a beard?	Grace Bedell.	hard	medium	S08_set3_a4
14	Abraham_Lincoln	When did the Gettysburg address argue that Ame...	1776	hard	hard	S08_set3_a4

Figure 3. Data exploration

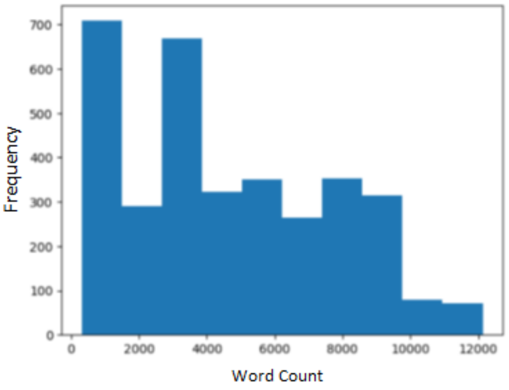


Figure 4. Article words length

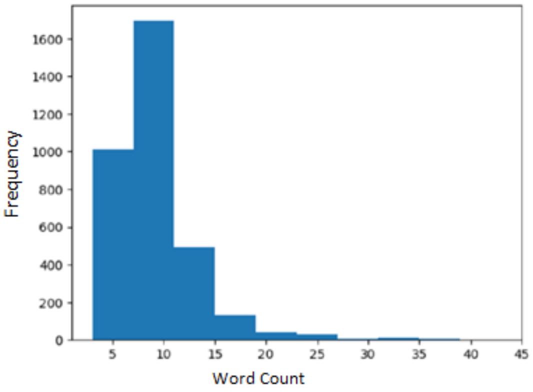
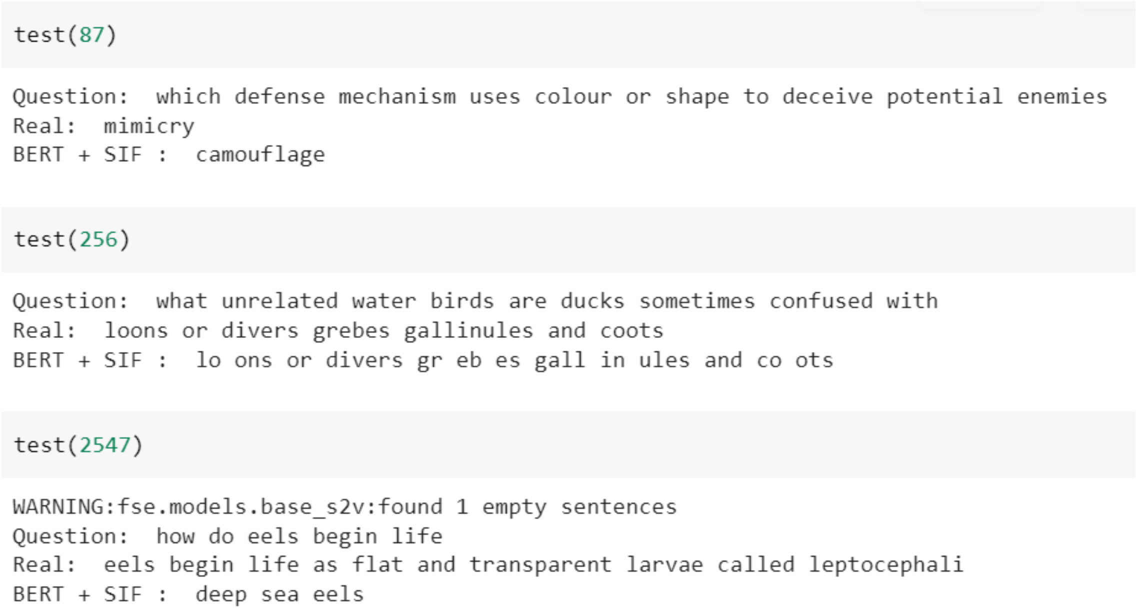


Figure 5. Question words length



```
test(87)

Question:  which defense mechanism uses colour or shape to deceive potential enemies
Real:  mimicry
BERT + SIF :  camouflage

test(256)

Question:  what unrelated water birds are ducks sometimes confused with
Real:  loons or divers grebes gallinules and coots
BERT + SIF :  lo ons or divers gr eb es gall in ules and co ots

test(2547)

WARNING:fse.models.base_s2v:found 1 empty sentences
Question:  how do eels begin life
Real:  eels begin life as flat and transparent larvae called leptocephali
BERT + SIF :  deep sea eels
```

Figure 6. BERT+uSIF illustration output

Table 1. Sample test cases

Question	Actual answer	BERT+SIF answer	Comments
Do all ducks quack	No	Most ducks other than female mall ards and domestic ducks do not quack	Correct with an elaborate answer
What are four species that are commonly referred to as kangaroos	The red kangaroo the eastern grey kangaroo the western grey Kangaroo and the antilopine kangaroo	The red kangaroo the anti-lo pine kangaroo and the eastern and western grey kangaroo	Partially correct
Why did Cleveland want to hide his cancer surgery from the public	Because of the financial depression in the country	Financial depression of the country Cleveland decided to have surgery performed on the tumor in secrecy to avoid further market panic	Correct with an elaborate answer
What established a trading post on the island in 1819	British East India Company	British east India company	same
How do elephants communicate over long distances	By producing and receiving low-frequency sound infrasound	Elephants are observed listening by putting trunks on the ground and carefully moving their very sensitive feet	Somewhat correct
What did Roosevelt do to improve his physical condition	Roosevelt took up exercise	To combat his poor physical condition his father compelled the young Roosevelt to take up exercise	Correct with an elaborate answer
Did Coolidge graduate from black river academy	Yes	Coolidge graduated from black river academy Vermont	Correct with an elaborate answer

Table 2. Performance comparison

	Accuracy (%)	F1 score (%)	EM score (%)
BERT	59.68	63.03	59.23
Longformer	-	83.35	79.92
BERT (SQuAD)	72.36	75.75	72.36
BERT+uSIF	81.34	85.53	54.28

4. CONCLUSION

Preliminary results show that the given approach is promising for longer text sequences as compared to the BERT base model which restricts up to 512 tokens. It can further be improved by selecting those chunks that might have answers and discarding the other chunks, and it might help in better performance and accuracy. It takes a lot of time to compute the answer. Performance can be improved further by combining it with the SIF method. When it comes to capturing the contextual connections between words and sentences in a document, the transformer-based model BERT excels. BERT can acquire a generalized model of language that can be honed for particular downstream tasks like QA by pre-training on a lot of data. Contrarily, LSTM is an RNN that excels at processing consecutive input. By independently encoding the input text and the question before merging them to produce a response, LSTM models have been used for tasks that require addressing questions.

Future research should focus on further developing the different versions of BERT and LSTM algorithms for QA tasks also focus should be given to improving the time and space complexity of these models. As the context grows, the time and space complexity of getting the answer also grows quadratically. The development of models that can better manage lengthy documents or numerous paragraphs of text is one possible direction for progress, as current models frequently battle with these types of inputs. Exploring how to integrate external knowledge sources into these models, such as knowledge graphs or domain-specific ontologies, to enhance their performance on particular kinds of queries is another field of study.

ACKNOWLEDGMENTS

We would like to express our gratitude to our institution, Vellore Institute of Technology (VIT), which has been of tremendous support to us in completing this proposal.

REFERENCES

[1] A. Otegi, I. S. Vicente, X. Saralegi, A. Peñas, B. Lozano, and E. Agirre, "Information retrieval and question answering: A case study on COVID-19 scientific literature," *Knowledge-Based Systems*, vol. 240, Mar. 2022, doi: 10.1016/j.knosys.2021.108072.

[2] Y. Huang, J. Liu, and C. Lv, "Chains-BERT: a high-performance semi-supervised and contrastive learning-based automatic question-and-answering model for agricultural scenarios," *Applied Sciences*, vol. 13, no. 5, Feb. 2023, doi: 10.3390/app13052924.

[3] H.-Y. Chan and M.-H. Tsai, "Question-answering dialogue system for emergency operations," *International Journal of Disaster Risk Reduction*, vol. 41, Dec. 2019, doi: 10.1016/j.ijdrr.2019.101313.

[4] Z. Xue, G. He, J. Liu, Z. Jiang, S. Zhao, and W. Lu, "Re-examining lexical and semantic attention: Dual-view graph convolutions enhanced BERT for academic paper rating," *Information Processing and Management*, vol. 60, no. 2, Mar. 2023, doi:

10.1016/j.ipm.2022.103216.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, Oct. 2018, [Online]. Available: http://arxiv.org/abs/1810.04805

[6] J. A. Alzubi, R. Jain, A. Singh, P. Parwekar, and M. Gupta, "COBERT: COVID-19 question answering system using BERT," *Arabian Journal for Science and Engineering*, vol. 48, no. 8, pp. 11003–11013, Aug. 2023, doi: 10.1007/s13369-021-05810-5.

[7] X. Lv, Z. Liu, Y. Zhao, G. Xu, and X. You, "HBert: a long text processing method based on BERT and hierarchical attention mechanisms," *International Journal on Semantic Web and Information Systems*, vol. 19, no. 1, pp. 1–14, May 2023, doi: 10.4018/IJSWIS.322769.

[8] A. Vaswani *et al.*, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, Jun. 2017, [Online]. Available: http://arxiv.org/abs/1706.03762

[9] D. Song, A. Vold, K. Madan, and F. Schilder, "Multi-label legal document classification: a deep learning-based approach with label-attention and domain-specific pre-training," *Information Systems*, vol. 106, May 2022, doi: 10.1016/j.is.2021.101718.

[10] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: the long-document transformer," *arXiv preprint arXiv:2004.05150*, Apr. 2020, doi: 10.48550/arXiv.2004.05150.

[11] J. Yin, "Research on question answering system based on BERT model," in *2022 3rd International Conference on Computer Vision, Image and Deep Learning and International Conference on Computer Engineering and Applications (CVIDL and ICCEA)*, IEEE, May 2022, pp. 68–71, doi: 10.1109/CVIDLICCEA56201.2022.9824408.

[12] M. Zaheer *et al.*, "Big bird: transformers for longer sequences," *Advances in neural information processing systems*, vol. 33, pp. 17283–17297, Jul. 2020, [Online]. Available: http://arxiv.org/abs/2007.14062

[13] Y. Liu, T. Hao, H. Liu, Y. Mu, H. Weng, and F. L. Wang, "OdeBERT: one-stage deep-supervised early-exiting BERT for fast inference in user intent classification," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 5, pp. 1–18, May 2023, doi: 10.1145/3587464.

[14] C. Anjun, X. Dingyuan, T. Liang, and S. Meining, "Development and application of an automatic question answering system for silkworm disease knowledge," in *2020 5th International Conference on Smart Grid and Electrical Automation (ICSGEA)*, IEEE, Jun. 2020, pp. 367–370, doi: 10.1109/ICSGEA51094.2020.00085.

[15] S. Zhao, F. You, W. Chang, T. Zhang, and M. Hu, "Augment BERT with average pooling layer for Chinese summary generation," *Journal of Intelligent and Fuzzy Systems*, vol. 42, no. 3, pp. 1859–1868, Feb. 2022, doi: 10.3233/JIFS-211229.

[16] N. Fujishiro, Y. Otaki, and S. Kawachi, "Accuracy of the sentence-BERT semantic search system for a Japanese database of closed medical malpractice claims," *Applied Sciences*, vol. 13, no. 6, Mar. 2023, doi: 10.3390/app13064051.

[17] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "SQuAD: 100,000+ questions for machine comprehension of text," Jun. 2016, [Online]. Available: http://arxiv.org/abs/1606.05250

[18] M. Joshi, E. Choi, D. Weld, and L. Zettlemoyer, "TriviaQA: a large scale distantly supervised challenge dataset for reading comprehension," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Stroudsburg, PA, USA, Canada: Association for Computational Linguistics, Jul. 2017, pp. 1601–1611, doi: 10.18653/v1/P17-1147.

[19] J. Zhou *et al.*, "Graph neural networks: a review of methods and applications," *AI Open*, vol. 1, pp. 57–81, 2020, doi: 10.1016/j.aiopen.2021.01.001.

[20] S. Shim, G. Chodwadia, K. Jain, C. Patel, E. Sorathia, and C. Choo, "Supervised question answering system for technical support," in *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, IEEE, Jan. 2018, pp. 216–220, doi: 10.1109/CCWC.2018.8301764.

[21] N. N. Khin and K. M. Soe, "Question answering based university chatbot using sequence to sequence model," in *2020 23rd Conference of the Oriental COCOSDA International Committee for the Co-ordination and Standardisation of Speech Databases and Assessment Techniques (O-COCOSDA)*, IEEE, Nov. 2020, pp. 55–59, doi: 10.1109/O-COCOSDA50338.2020.9295021.

[22] Z. Bo, W. Haowen, J. Longquan, Y. Shuhan, and L. Meizi, "A novel bidirectional LSTM and attention mechanism based neural network for answer selection in community question answering," *Computers, Materials and Continua*, vol. 62, no. 3, pp. 1273–1288, 2020, doi: 10.32604/cmc.2020.07269.

[23] A. F. Hanifah and R. Kusumaningrum, "Non-factoid answer selection in indonesian science question answering system using long short-term memory (LSTM)," *Procedia Computer Science*, vol. 179, pp. 736–746, 2021, doi: 10.1016/j.procs.2021.01.062.

[24] L. Enamoto, A. R. A. S. Santos, R. Maia, L. Weigang, and G. P. R. Filho, "Multi-label legal text classification with BiLSTM and attention," *International Journal of Computer Applications in Technology*, vol. 68, no. 4, p. 369, 2022, doi: 10.1504/IJCAT.2022.125186.

[25] C. Park *et al.*, "S 2 -Net: Machine reading comprehension with SRU-based self-matching networks," *ETRI Journal*, vol. 41, no. 3, pp. 371–382, Jun. 2019, doi: 10.4218/etrij.2017-0279.

[26] S. Arora, Y. Liang, and T. Ma, "A simple but tough-to-beat baseline for sentence embeddings," in *International Conference on Learning Representations*, Jan. 2017, pp. 1–16.

[27] A. Onan, "Hierarchical graph-based text classification framework with contextual node embedding and BERT-based dynamic fusion," *Journal of King Saud University - Computer and Information Sciences*, vol. 35, no. 7, Jul. 2023, doi: 10.1016/j.jksuci.2023.101610.

BIOGRAPHIES OF AUTHORS

Dr. Vijayan Ramaraj is working as a professor at the School of Computer Science Engineering and Information Systems (SCORE), Vellore Institute of Technology (VIT), Vellore, Tamil Nadu, India. He is a life member of the Computer Society of India (CSI). He has produced several national and international research articles in reputed journals and conferences. His research interest involves web technologies, wireless networks, ad hoc networks, computer networks, cloud computing, and data analytics. He can be contacted at email: rvijayan@vit.ac.in.

Dr. Mareeswari Venkatachala Appa Swamy is working as an assistant professor (senior) at the SCORE, VIT, Vellore, India. She is a life member of the CSI. She received her Ph.D. in the area of web service and has produced several national and international articles in reputed journals and conferences. Her areas of interest include programming, web services, image processing, artificial intelligence, ad-hoc networks, and data analytics. She can be contacted at email: vmareeswari@vit.ac.in.

Dr. Ephzibah Evan Prince received her Ph.D. from Mother Teresa Women's University, Kodaikanal, India. She is currently working as an associate professor at SCORE, VIT, Vellore, India. She has over 20 years of academic and research experience. Her research area includes artificial intelligence, soft computing techniques, blockchain technology, classification, clustering, and prediction. She can be contacted at email: ep.ephzibah@vit.ac.in.

Chandhan Kumar was a student of master of computer application in SCORE, VIT, Vellore, India. He is currently working as Software Engineer 1 B in BA Continuum Pvt Ltd, Bank of America, Hyderabad, Telangana, India. He can be contacted at email: ckp1606@gmail.com.